(1)

AD-A164 021

ESTIMATION OF ICBM PERFORMANCE PARAMETERS

THESIS

Ronald A. Worley
Captain, USAF

AFIT/GA/AA/85D-10

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

DTIC
ELECTE
FEB 12 1986
S D
B

DTIC FILE COPY

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

86 2 12 071

AFIT/GA/AA/85D-10

ESTIMATION OF ICBM PERFORMANCE PARAMETERS

THESIS

Ronald A. Worley
Captain, USAF

AFIT/GA/AA/85D-10

DTIC
ELECTE
FEB 1 2 1986
S D
B

Approved for public release; distribution unlimited

AFIT/GA/AA/85D-10

ESTIMATION OF ICBM PERFORMANCE PARAMETERS

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Astronautical Engineering

Ronald A. Worley, B. S.

Captain, USAF

December 1985

Approved for public release; distribution unlimited

ACKNOWLEDGEMENTS

Upon completing my thesis, I would like to express my deepest gratitude to my thesis advisor, Dr. Wiesel. His dedication and invaluable assistance helped make this project a true learning experience.

I would also like to thank my wife, Linda, without whose understanding and patience would have made my task much more difficult.

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
Dist
A-1

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF Tables

ABSTRACT

The estimation of launch vehicle performance parameters was explored through the use of a Bayes Filter. The main emphasis was to devise the means to detect a staging event, estimate the staging time and next stage vehicle parameters, and reenter the main Bayes Filter to process subsequent stage observation data. The state model consisted of the vehicle position and velocity vectors, the exhaust velocity,and the mass ratio. The results indicated that the staging event could successfully be detected by comparing the position of the vehicle as represented by the observation data and the position as represented by the numerical integrator. The exhaust velocity and mass ratio of the next stage could not be estimated independently. The staging estimater state model was then altered to estimate the product of the exhaust velocity and mass ratio. The problems encountered reentering the main Bayes Filter were identical to the ones the staging estimator had. It was then determined that there was a possible observability problem with the main algorithm. It was recommended that the main state vector be altered to include the product of the exhaust velocity and mass ratio rather than thier independent estimation.

I. INTRODUCTION

The determination of launch vehicle performance parameters was last investigated by Capt. Vallado (reference 4) with marginal results. The difficulties encountered were a result of a staging event occurring during the beginning or middle of a Bayes Filter (reference 5) segment and the dynamics not modeling the problem adequately once the staging event occurred. The specific task that will be examined in this research will be an attempt to "capture" the staging event and remodel the dynamics once the staging event has been detected by the computer algorithm. The importance of gaining knowledge of foreign technology, as stated by Capt. Vallado (reference 4), would aid the United States in planning strategic defense policies and determine where emphasis should be placed for future research and development.

The problem we will be addressing will involve the detection of the launch of a missile and the subsequent track of that vehicle by either an orbiting sensor or land sensor. The data obtained will then be used to determine the launch vehicle parameters to aid in identification of the type of vehicle launched or the classification of a new type vehicle, as the case may be.

Figure 1 shows the geometry involved in the problem of a land based sensor tracking the vehicle. In "real life" the vehicle will not be tracked until it appears over the horizon or in the field of view of the sensor. This, in all

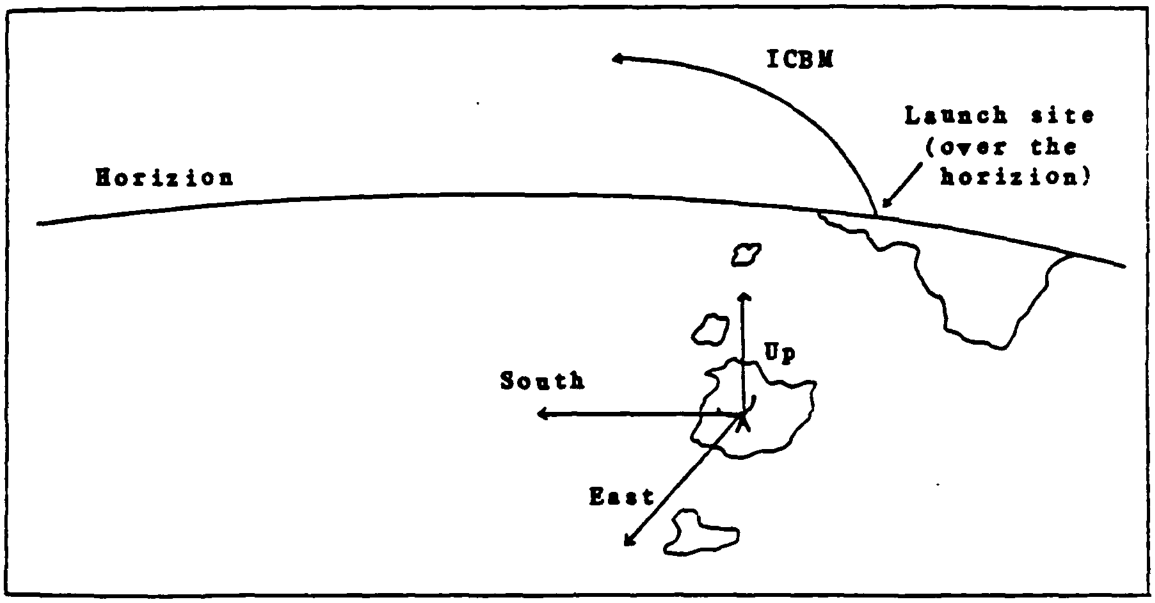

Figure 1. Land Based Sensor Geometry

cases, will be some time period after launch has occurred. For this paper it will be assumed that the data is available for the entire trajectory of the vehicle and its point of origin is known.

Figure 2 shows the geometry involved in the problem of an orbiting sensor tracking the vehicle. The problems associated with an orbiting sensor is that it must identify the vehicle among the "ground clutter" as it is looking down on the launch trajectory. With recent developments in radar technology, this problem is becoming less of a concern. The problem being investigated could be readily applied to recent developments in the Strategic Defense Initiative. The detection and subsequent threat assessment of a launch vehicle of unknown origin is invaluable.

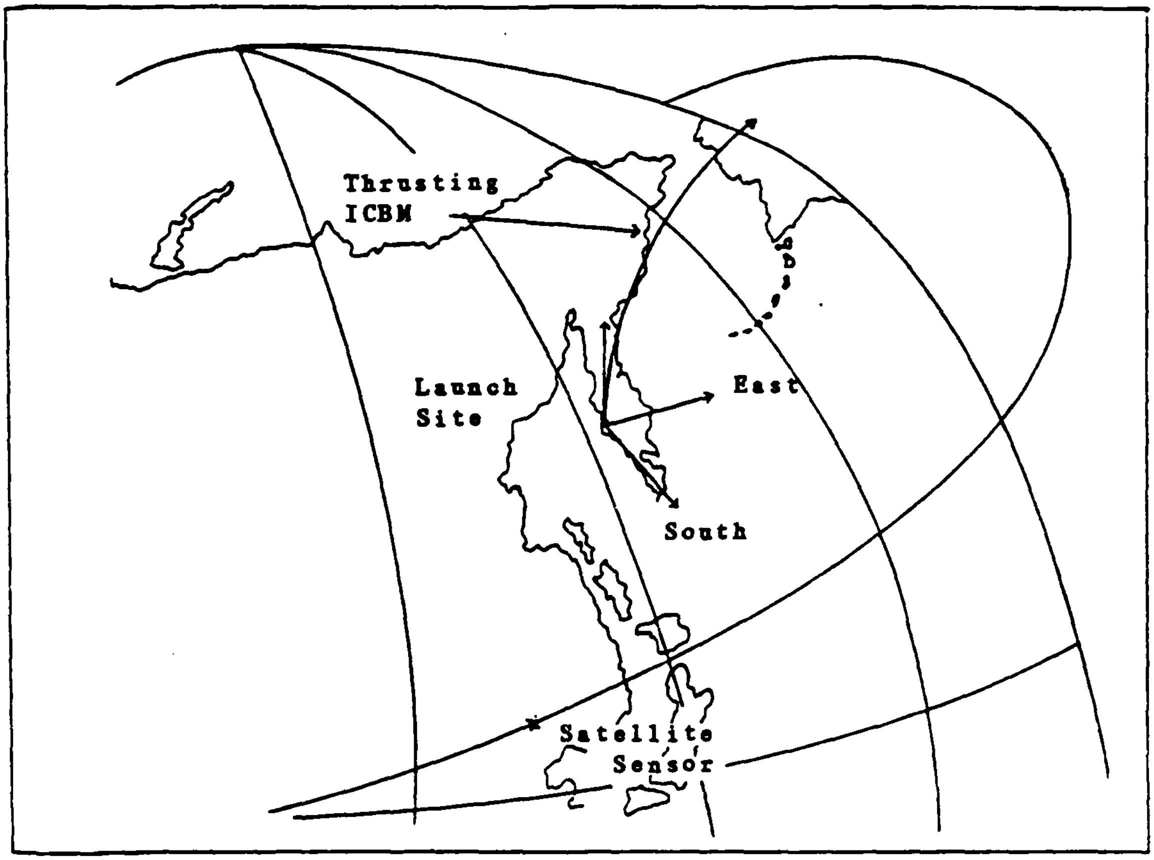

Figure 2. Orbiting Sensor Geometry

Work done in this area prior to Capt. Vallado's thesis included the determination of position and velocity vectors and vehicle acceleration using a 7-state filter from elevation and azimuth data (reference 2). Capt. Gross's thesis made the first use of a Bayes Filter to determine exhaust velocity, vehicle mass, and vehicle acceleration. This resulted in good results for vehicle acceleration and poor results for exhaust velocity and vehicle mass. As was point-

ed out by Capt. Vallado, the complexity of the data and the observation relationships were partly responsible for the problems encountered.

II. PROBLEM DYNAMICS

EQUATIONS OF MOTION

For the problem being investigated, the equations of motion must be developed. The Bayes Filter (reference 5) algorithm will use the numerical integration of these equations to estimate the data that the radar sites would be observing. In developing these equations, a spherical earth model and the two-body equations of motion (reference 1) are considered. This seems to be a reasonable assumption as all trajectories considered will be "near Earth" types and all other gravitational force considerations will be small in comparison.

In general, the two-body equation of motion is:

$$\ddot{\bar{r}} + \bar{r}\mu/r^3 = 0 \qquad (2\text{-}1)$$

where

$\ddot{\bar{r}}$ = vehicle acceleration
$\bar{r}$ = radius vector from center of Earth to vehicle
r = radius vector magnitude
μ = gravitational parameter defined by:

$$\mu = GM \qquad (2\text{-}2)$$

where

G = Universal Gravitational Constant
M = Mass of the Earth

Only gravitational and thrust forces are considered in this paper as they are assumed to be several orders of magnitude larger than other forces that could be present (i.e. drag,

solar radiation, etc.).

The acceleration due to thrust can be derived by recalling the equations of motion of the rocket (reference 6) as illustrated in Figure 3. By following the expended particle

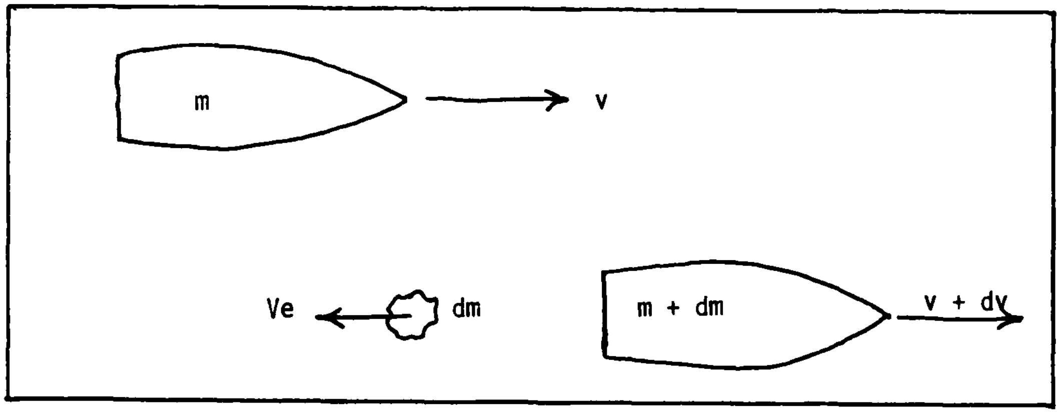

Figure 3. Rocket Thrust

of fuel, the equation of motion of the rocket can be developed. This is a closed system of particles, and therefore, must conserve linear momentum. To start, the linear momentum is mv, where m is the mass of the vehicle and v is the velocity of the vehicle. After a time dt, an incremental particle of fuel (dm) is expelled by the rocket at a velocity V_e with respect to the rocket, as shown in Figure 3. The incremental particle of fuel has a velocity of $V_e + v$ with respect to the inertial frame. The rocket has lost mass dm (interpreted as a negative quantity) but has gained velocity dv. Therefore, the conservation of momentum can be expressed as:

$$(m + dm)(v + dv) - dm(v + V_e) = mv \qquad (2\text{-}3)$$

Expanding Equation (2-3) and neglecting higher order terms:

$$m\,dv - dm\,V_e = 0 \qquad (2\text{-}4)$$

Dividing Equation (2-4) by dt and taking the limit leaves:

$$\dot{m}\,V_e = m\,a_{thrust} \qquad (2\text{-}5)$$

where

V_e = exhaust velocity
$\dot{m}V_e$ = thrust

Noting that the instantaneous mass term (m) can be written to reflect changing value with respect to time, Equation (2-5) can be rewritten in the direction of the velocity vector as:

$$\bar{a}_{thrust} = \frac{\dot{m}\,V_e}{(m_o - \dot{m}t)} \cdot \frac{\bar{v}}{|\bar{v}|} \qquad (2\text{-}6)$$

where

$\bar{a}_{thrust}$ = vehicle acceleration due to thrust along the velocity vector
$\dot{m}$ = mass flow rate
m_o = initial mass
t = time
V_e = Vehicle exhaust velocity

This particular equation is sufficient if the problem involved singly-staged vehicles only, as presented in Capt. Vallado's thesis (reference 4). In order to present the problem in a more general form, it must be modified to allow for multi-staged vehicles. Therefore, Equation (2-6) must be rewritten as follows:

$$\bar{a} = \frac{V_e \dot{m}}{[\, m_o - \dot{m}\,(t - t_{stage})]} \cdot \frac{\bar{v}}{|\bar{v}|} \qquad (2\text{-}7)$$

where

t_{stage} = the time the staging event takes place

Since the absolute masses are not observable from the trajectory data, let $M = \dot{m} \,/\, m_o$, and:

$$\bar{a} = \frac{V_e \; M}{[1 - M(t - t_{stage})]} \cdot \frac{\bar{v}}{|\bar{v}|} \qquad (2\text{-}8)$$

By recalling Newton's Law (the mass times acceleration equals the sum of the forces), the total vehicle acceleration is obtained by combining Equations (2-1) and (2-8) as follows:

$$\ddot{\bar{r}} = - \frac{\bar{r}\,\mu}{r^3} + \frac{V_e \; M}{[1 - M(t - t_{stage})]} \cdot \frac{\bar{v}}{|\bar{v}|} \qquad (2\text{-}9)$$

where $\ddot{\bar{r}}$ denotes the total vehicle acceleration.

Observation Relationships

Two cases were considered while developing this algorithm. The first case was of an orbiting sensor observing a launch vehicle trajectory from above. The next case involved a ground radar site tracking a launch vehicle above the horizon. The observation relationships for both cases are the same with the exception of the position vectors of the radar sites.

The position vector of the ground radar site can be de-

termined once the latitude, longitude, elevation, and universal time are known. For these values we begin by calculating the local sidereal time for the site (reference 1):

$$\theta = \theta_g + \lambda_e \qquad (2\text{-}10)$$

where

θ = local sidereal time in degrees
θ_g = Greenwich sidereal time in degrees
λ_e = longitude of the site in degrees

The Greenwich sidereal time is calculated as follows:

$$\theta_g = \theta_{go} + 1.0027379093\ (t - t_o)\ 2\pi \qquad (2\text{-}11)$$

where

θ_g = Greenwich sidereal time in degrees
θ_{go} = value in degrees on 1 Jan that year
$(t - t_o)$ = time in days past initial time
1.0027379093 days of mean sidereal time = 1 day of mean solar time

The site position vector can be calculated as follows:

$$\bar{r}s = \begin{bmatrix} h \cos (L) \cos (\theta) \\ h \cos (L) \sin (\theta) \\ h \sin (L) \end{bmatrix} \qquad (2\text{-}12)$$

where

h = distance from center of earth to site
$\bar{r}s$ = site vector
L = latitude of site
θ = local sidereal time

The coordinate system for the land based sensor is shown in Figure 4.

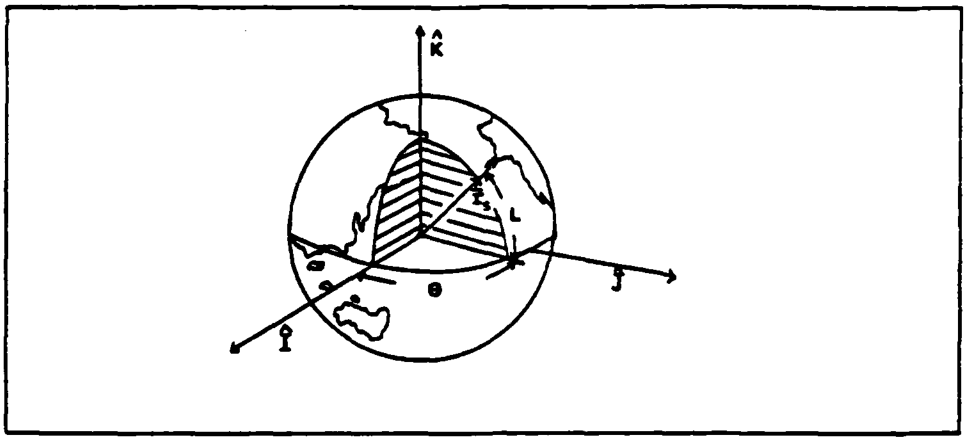

Figure 4 Land Based Sensor Coordinate System

The orbiting sensor is a bit more involved as it requires the orbit of the sensor be known. The semi-major axis (a), eccentricity (e), inclination (i), longitude of ascending node (Ω), and the argument of periapsis (ω) are input (reference 1) so position and velocity vectors can be determined. The procedure requires the determination of the mean motion and then the mean anomaly. With these values, $\bar{r}$ and $\bar{v}$ of the orbiting sensor can be calculated as follows:

$$\bar{r} = r \cos \nu \; \hat{P} \; + \; r \sin \nu \; \hat{Q}$$

and

$$\bar{v} = [\ \mu \ / \ p \ ]^{1/2} \ [\ - \sin \nu \hat{P} \ + (\ e + \cos \nu \ ) \hat{Q} \ ]$$

where

ν = true anomoly
$\hat{P}$,$\hat{Q}$ = unit vectors in the Perifocal Coordinate System (PQW, reference 1)
p = semi-latus rectum

For a more detailed description, see Appendix A of Vallado's

thesis (reference 4).

With the site vector known in both cases, range, azimuth, elevation, and local coordinate system can be determined. The Topocentric-Horizon Coordinate System (SEZ), represented in Figure 5, is used to represent range, azimuth, and elevation data of the launch vehicle (reference 1). Notice that the azimuth angle is measured from the South rather than the North as is done by most radar sites. This is done to simplify later computations.

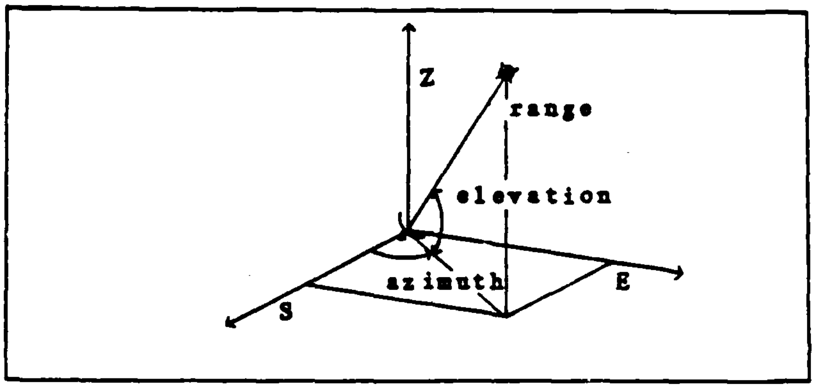

Figure 5. Radar Site Geometry

Since the state vector and all input data are in the Geocentric-Equatorial Coordinate System (IJK) (reference 1), an orthogonal set of unit vectors in the SEZ frame must be developed to be used as a transformation matrix. Figure 6 will aid in determining the transformation matrix.

The local vertical unit vector ($\hat{Z}$) is derived as:

$$\hat{Z} = \bar{r}_s / |\bar{r}_s| \tag{2-13}$$

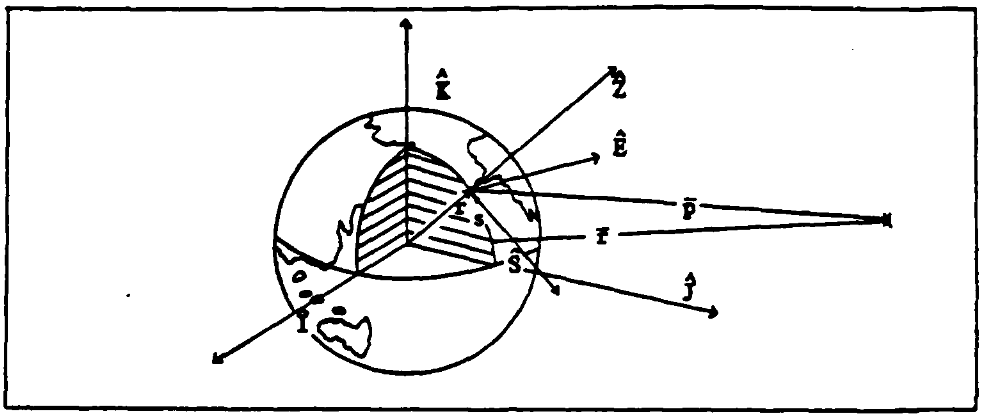

Figure 6 Observation Geometry

The East unit vector is defined as:

$$\hat{E} = \hat{k} \times \hat{Z} / |\hat{k} \times \hat{Z}| \tag{2-14}$$

While the South unit vector is defined as:

$$\hat{S} = \hat{E} \times \hat{Z} / |\hat{E} \times \hat{Z}| \tag{2-15}$$

The transformation matrix can then be shown as:

$$(\cdot)_{IJK} = [\, \hat{S} \mid \hat{E} \mid \hat{Z} \,] \quad (\cdot)_{SEZ} \tag{2-16}$$

where

$(\cdot)_{IJK}$ = an arbitrary vector in the IJK frame
$(\cdot)_{SEZ}$ = an arbitrary vector in the SEZ frame

$[\, \hat{S} \mid \hat{E} \mid \hat{Z} \,]$ = orthogonal basis vector , 3 x 3 matrix consisting of Equations (2-13), (2-14), and (2-15)

Since the inverse of an orthogonal transformation matrix is

the transpose, it can also be stated that:

$$(\cdot)_{SEZ} = \begin{bmatrix} \hat{S} \\ \hline \hat{E} \\ \hline \hat{Z} \end{bmatrix} (\cdot)_{IJK} \qquad (2\text{-}17)$$

With the proper transformation matrices developed, it is now possible to compute the range, azimuth, and elevation of the launch vehicle. The range is be defined as:

$$\rho_{IJK} = \bar{r}_{IJK} - \bar{r}_{SIJK} \qquad (2\text{-}18)$$

and transforming to the SEZ frame:

$$\rho_{SEZ} = \begin{bmatrix} \hat{S} \\ \hline \hat{E} \\ \hline \hat{Z} \end{bmatrix} \rho_{IJK} \qquad (2\text{-}19)$$

Then the data is assembled as follows:

$$\begin{aligned} \text{range} &= \rho \\ \text{azimuth} &= \tan^{-1} (y / x) \qquad (2\text{-}20) \\ \text{elevation} &= \tan^{-1} (z / (x^2 + y^2)^{1/2}) \end{aligned}$$

where

x, y, z = components of the position vector in the IJK frame

The method for finding the unit vectors for both sensor types is the same. The only difference is the calculation of the initial site vector.

Truth Model Data

Programming the equations of motion and numerically integrating them provided the truth model data. Appendix A lists the truth model program and all subroutines needed to generate the simulated radar data needed to test the estimation algorithm. To generate data for a particular launch vehicle it was necessary to obtain information of current launch vehicles in the present U. S. inventory. Specifically needed was V_e and M ($\dot{m}$ / m_o) which were obtained from reference 3. Recalling from basic propulsion (reference 6):

$$\dot{m} = F / V_e \quad \text{and} \quad V_e = I_{sp}\, g \quad \text{and} \quad M = \dot{m} / m_o$$

only I_{sp}, m_o, and F need to be specified. The values used in the truth model are seen in Table 1.

Table 1 Launch Vehicle Data

Titan 34 D				
STAGE	TIME(sec)	ISP(sec)	THRUST(lb_f)	MASS$_o$ (lb_m)
1	0.0	301.6	531,250.0	410,028.0
2	165.0	318.0	100,700.0	102,028.0
3	375.0	295.0	42,200.0	24,028.0
4	520.4	cutoff	0.0	2,628.0

By numerically integrating Equation (2-9) and using Table 1, the observation data (truth model), Equation (2-20), was generated.

Next, the truth model was generated to include noisy data, to simulate "real-life" data. The method to obtain noisy data was to have the data files written to include random errors in range, azimuth, and elevation. The method involved the assumption that the errors would occur randomly as per a Gaussian distribution, as presented by Vallado's thesis (reference 4). The difference between measurements of noisy data and "perfect"data is defined by:

$$d \begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix} = \text{Gau} \ \sigma \begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix} \qquad (2\text{-}21)$$

where

Gau represents a Gaussian function whose mean = 0 and standard deviation is ± 1

σ is the defined accuracy of the three measurements

The accuracies of the range, azimuth, and elevation measurements are specified as:

$$\sigma \begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix} = \begin{bmatrix} .00001 \text{ DU} \\ .001 \text{ deg} \\ .001 \text{ deg} \end{bmatrix} \sim 64 \text{ m} \qquad (2\text{-}22)$$

Then using partial derivatives, the deviations are:

$$\delta \begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix} = \text{Gau} \begin{bmatrix} .00001 \\ .001 \\ .001 \end{bmatrix} \delta \begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix} \qquad (2\text{-}23)$$

The partial of range, azimuth, and elevation then becomes an identity matrix since it is assumed that the random errors associated with the three measurements are independent of

each other. The noisy data is then formed as:

$$\begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix}_{\text{noisy}} = \begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix}_{\text{perfect}} + \delta \begin{pmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{pmatrix} \quad (2\text{-}24)$$

The filters will be developed next to implement the dynamics formulated in this chapter.

III. FILTER DEVELOPMENT

The estimation routine will be required to process sensor data (azimuth, elevation, range, and time) and determine positional, velocity, and launch vehicle stage data. A sequential estimator, Bayes Filter (reference 6), will be used to facilitate the detection of the staging event.

Matrix Equations

Specification of the state vector (reference 4) will be done using what was developed in Chapter 2. It is defined as:

$$\bar{x} = \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ V_e \\ M \end{bmatrix} \tag{3-1}$$

where

x, y, z = components of position
$\dot{x}, \dot{y}, \dot{z}$ = components of velocity
V_e = exhaust velocity of stage
M = mass ratio of $\dot{m} / m_o$

Since the two-body equations of motion are nonlinear, the state must be moved to the next observation time using a numerical integrator (Haming, a fourth-order predictor-correc-

tor was chosen, reference 7). The equations of motion are

$$d/dt(\overline{x}(t)) = \overline{F}(\overline{x}(t),t) \qquad (3\text{-}2)$$

where x(t) is the state vector at each time. This is just another expression for Equation (2-9). The $\overline{F}$ vector is found to be

$$\overline{F}(\overline{x}(t),t) = \begin{bmatrix} x \\ y \\ z \\ x \\ y \\ z \\ V_e \\ M \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ -x \;/\; r^3 + a\,\dot{x} \;/\; v \\ -y \;/\; r^3 + a\,\dot{y} \;/\; v \\ -z \;/\; r^3 + a\,\dot{z} \;/\; v \\ 0 \\ 0 \end{bmatrix} \qquad (3\text{-}3)$$

where

$$a = \frac{V_e \; M}{[\; 1 - M(t - t_{stage}) \;]}$$

and V_e and M are assumed to be constant for a particular stage.

Now that the relations involving the equations of motion have been formulated, the relations needed to correct the estimate of the state vector using the input data must be formulated. To estimate the state, a nominal trajectory is assumed as a function of time ($\overline{x}(t)$), with initial conditions. The true trajectory can be written as:

$$\overline{x}(t) = \overline{x_o}(t) + \delta\,\overline{x}(t) \qquad (3\text{-}4)$$

where

$\overline{x}(t)$ = true trajectory

$\delta\,\overline{x}(t)$ = difference between true and nominal trajectory

$\overline{x_o}(t)$ = nominal trajectory

Differentiating the true trajectory relation yields:

$$\dot{\overline{x}}(t) = \dot{\overline{x_o}}(t) + \delta\,\dot{\overline{x}}(t) \qquad (3\text{-}5)$$

and modifying Equation (3-2) produces:

$$\dot{\overline{x_o}}(t) + \delta\,\dot{\overline{x}}(t) = \overline{F}(\overline{x_o}(t) + \delta\,\overline{x}(t), t) \qquad (3\text{-}6)$$

To solve this equation, we expand the right hand side using a Taylor's Series expansion to obtain:

$$\dot{\overline{x_o}}(t) + \delta\,\dot{\overline{x}}(t) = \overline{F}(\overline{x_o}(t), t) + \partial\overline{F}/\partial t\Big|_{\overline{x}o(t)} \;\delta\,\overline{x}(t) + \frac{1}{2}\;\nabla_x\,(\nabla_x\,\delta\,)\Big|_{\overline{x}o(t)} \;(\delta\,\overline{x}(t)\,)^2 + \text{H.O.T.} \qquad (3\text{-}7)$$

Now assuming that $\delta\,\overline{x}(t)$ is zero, Equation (3-7) becomes:

$$\dot{\overline{x_o}}(t) = \overline{F}(\overline{x_o}(t), t) \qquad (3\text{-}8)$$

Ignoring higher order terms in Equation (3-7) and subtracting Equation (3-8) leaves:

$$\delta\,\dot{\overline{x}}(t) = A(t)\Big|_{\overline{x}o(t)} \;\delta\,\overline{x}(t) \qquad (3\text{-}9)$$

where $A(t) = \partial\overline{F}(t)/\partial\overline{x}$ (the A matrix is derived in the Appendix B)

Recalling Equation (3-4), we may specify the following rela-

tion:

$$\bar{x}(t) = \phi(t,t_o)\ \delta\bar{x}(t_o) \tag{3-10}$$

where $\phi(t,t_o) \equiv \partial\bar{x}(t)/\partial\bar{x}(t_o)\Big|_{\bar{x}_o(t)}$ is a square matrix, the state transition matrix. When evaluated along a known solution, the state transition matrix is a function of the start and stop times only. It calculated from

$$\dot{\phi}(t,t_o) = A(t)\Big|_{\bar{x}_o(t)}\ \phi(t,t_o) \tag{3-11}$$

and the state transition matrix initial condition is prescribed by

$$\phi(t_o,t_o) = I \tag{3-12}$$

where I = identity matrix.

The formulation of the equations responsible for the calculation of the state vector, its propagation through time, and estimation of the errors from the true trajectory have been completed. The next step is to process the data coming from the sensors. The data will usually be a nonlinear function of the state vector at the current time (t_i) and of the observation geometry.

The predicted data for each observation is given by:

$$\bar{z}(t_i) = \bar{G}(\bar{x}(t_i),t_i) \tag{3-13}$$

By evaluating this at the initial time we are left with the initial conditions as:

$$\overline{z_o}(t_i) = \overline{G}(\overline{x_o}(t_i), t_i) \qquad (3\text{-}14)$$

This equation can be linearized as we did with Equation (3-2). Knowing that there will be a difference from the nominal trajectory, Equation (3-13) can be rewritten as

$$\overline{z}(t_i) = \overline{G}(\overline{x_o}(t_i) + \delta\,\overline{x}(t_i), t_i) \qquad (3\text{-}15)$$

where $\delta x(t_i)$ is the difference perturbation from true data. This can be expanded as was done in Equation (3-6) and yields:

$$\overline{z}(t_i) = \overline{G}(\overline{x_o}(t_i), t_i) + \partial\overline{G}(x_o(t_i), t_i)/\partial\,\overline{x}(t_i)\Big|_{x_o(t_i)} \delta\,\overline{x}(t_i) + \text{H.O.T.} \qquad (3\text{-}16)$$

Subtracting this 'true' relation from the calculated relation and ignoring higher order terms produces:

$$\begin{aligned} \overline{r}(t_i) &= \overline{z}(t_i) - \overline{z_o}(t_i) \\ &= \partial\overline{G}/\partial\,\overline{x}\Big|_{x_o(t_i)} \delta\,\overline{x}(t_i) \\ &= H(\overline{x_o}(t_i), t_i)\ \delta\,\overline{x}(t_i) \end{aligned} \qquad (3\text{-}17)$$

In previous chapters the observation relationships were developed. The data vector G consists of:

$$[G] = \begin{bmatrix} \text{range} \\ \text{azimuth} \\ \text{elevation} \end{bmatrix} \qquad (3\text{-}18)$$

The H matrix is defined as:

$$[H] = \partial\overline{G}/\partial\overline{x} \tag{3-19}$$

Using the observation relationships that were developed in Chapter 1:

$$\begin{aligned} \text{range} &= (x^2 + y^2 + z^2)^{1/2} \\ \text{azimuth} &= \tan^{-1} (y / x) \\ \text{elevation} &= \tan^{-1} [z / (x^2 + y^2)^{1/2}] \end{aligned} \tag{3-20}$$

where

x,y,z = positional components in the IJK frame

Since only x, y, and z appear in the H matrix, the first 3 x 3 block will be the only portion that is not zero. Therefore the first 3 x 3 block of the H matrix is:

$$[H] = \begin{bmatrix} \dfrac{x}{(x^2+y^2+z^2)^{1/2}} & \dfrac{y}{(x^2+y^2+z^2)^{1/2}} & \dfrac{z}{(x^2+y^2+z^2)^{1/2}} \\[2ex] \dfrac{-y/x^2}{1 + (y/x)^2} & \dfrac{1/x}{1 + (y/x)^2} & 0 \\[2ex] \dfrac{-xz/(x^2+y^2)^{3/2}}{1 + z^2/(x^2+y^2)} & \dfrac{-yz/(x^2+y^2)^{3/2}}{1 + z^2/(x^2+y^2)} & \dfrac{1/(x^2+y^2)^{1/2}}{1 + z^2/(x^2+y^2)} \end{bmatrix} \tag{3-21}$$

Since the estimate of the state is at an epoch time, t_o, from measurements taken at different observation times, t_i, the residuals are moved to a single epoch time. Using the state transition matrix as before:

$$\begin{aligned} \overline{r}(t_i) &= H(\overline{x_o}(t_i),t_i)\,\phi(t_i,t_o)\;\delta\,\overline{x}(t_o) \\ &= T(t_i)\;\delta\,\overline{x}(t_o) \end{aligned} \tag{3-22}$$

where $T(t_i)$ is defined as $H(\overline{x_o}(t_i),t_i)\,\phi(t_i,t_o)$.

Now that the required matrices have been defined, the estimation routines can be developed.

Nonlinear Least Squares

From what has already been developed, the state vector is given by:

$$\dot{\bar{x}} = \bar{F}(\bar{x},t) \tag{3-23}$$

with deviation of the state vector as:

$$\delta\bar{x}(t) = \phi(t,t_o)\ \delta\bar{x}(t_o) \tag{3-24}$$

The observation relationships were developed as the G matrix, and residual data was:

$$\bar{r}(t_i) = T(t_i)\ \delta\bar{x}(t_o) \tag{3-25}$$

The sensor data will not produce perfect data and the covariance matrix Q tells how accurate the range, azimuth, and elevation measurements are. The residual vector, including this error, can be shown as:

$$\bar{r}(t_i) = T(t_i)\ \delta\bar{x}(t_o) + \bar{e}(t_i) \tag{3-26}$$

where

$\bar{e}(t_i)$ = acrual error associated with the observation data

Solving Equation (3-26) for the error:

$$\bar{e}(t_i) = \bar{r}(t_i) - T(t_i)\ \delta\bar{x}(t_o) \tag{3-27}$$

For this research, it is assumed that random errors in range, azimuth, and elevation are uncorrelated. The covariance ma-

trix (Q) is then defined, containing information as to the accuracy of the measurements.

Using Gaussian error statistics, the probability density function for the error vector is:

$$P(e) = (2\pi)^{-n/2} \left|Q\right|^{-1/2} \exp(-1/2\ S) \qquad (3\text{-}28)$$

where

n = number of measurements
Q = data covariance matrix
S = $\overline{e}^T Q^{-1} \overline{e}$ (a scalar) weighted least squares function

Using the principle of maximum likelihood (reference 5), S (volume of error ellipsoid) is minimized to make P a maximum. Therefore:

$$\partial S/\partial x = \partial(\overline{e}^T Q^{-1} \overline{e}) / \partial x \qquad (3\text{-}29)$$

Now substituting Equation (3-27) into S leaves:

$$\begin{aligned} S &= (\overline{r} - T\ \delta\overline{x})^T\ Q^{-1}\ (\overline{r} - T\ \delta\overline{x}) \\ &= \overline{r}^T\ Q^{-1}\ \overline{r} - \overline{r}^T\ Q^{-1}\ T\ \delta\overline{x} - \delta\overline{x}\ T^T\ Q^{-1}\ \overline{r} \\ &\quad + \delta\overline{x}^T\ T^T\ Q^{-1}\ T\ \delta\overline{x} \end{aligned} \qquad (3\text{-}30)$$

Note that the functional dependence on time has been left out to help in the clarity. Equation (3-29) becomes:

$$\begin{aligned} 0 = &- (\overline{r}^T\ Q^{-1}\ T)^T - T^T\ Q^{-1}\ \overline{r} + (\delta\overline{x}^T\ T^T\ Q^{-1}\ T)^T \\ &+ T^T\ Q^{-1}\ T\ \delta\overline{x} \end{aligned} \qquad (3\text{-}31)$$

We then solve Equation (3-31) for $\delta\overline{x}$:

$$\delta\overline{x} = (T^T\ Q^{-1}\ T)^{-1}\ T^T\ Q^{-1}\ \overline{r} \qquad (3\text{-}32)$$

This result is valid when the reference trajectory and the actual trajectory are very close and the inverse of $T^T Q^{-1} T$ exists. The requirement that $T^T Q^{-1} T$ have an inverse is called the observability condition (reference 5).

Now that we have an estimate, we must now assess the quality of the estimate. We need to calculate the covariance of the estimate as:

$$P_x(t) = E(\delta\overline{x}(t) \quad \delta\overline{x}(t)^T) \tag{3-33}$$

Noting that δx can be written as

$$\delta\overline{x} = W\,\overline{r} \tag{3-34}$$

and substituting it into Equation (3-33)and evaluating at t_o, leaves:

$$Px(t_o) = W\,E(\overline{r}\;\overline{r}^T)\,W^T \tag{3-35}$$

This assumes that W is deterministic and can therefore be brought outside of the expectation operator. Recalling that $E(\overline{r}\;\overline{r}^T)$ is defined as the covariance matrix Q where $\overline{r}$ is the the zero mean, it is shown:

$$P_x(t_o) = W\,Q\,W^T \tag{3-36}$$

Expanding this with the definition of W:

$$\begin{aligned} P_x(t_o) &= (T^T\;Q^{-1}\;T)^{-1}\;T^T\;Q^{-1}\;Q\;[(T^T\;Q^{-1}\;T)^{-1}\;T^T\;Q^{-1}]^T \\ &= (T^T\;Q^{-1}\;T)^{-1}\;T^T\;Q^{-1}\;T\;(T^T\;Q^{-1}\;T)^{-1} \\ &= (T^T\;Q^{-1}\;T)^{-1} \end{aligned} \tag{3-37}$$

The final step is to define when the estimator has reached convergence. Ideally $\delta\bar{x}$ will reach zero, but it is sufficient to stop the iteration when the state corrections are all less than the square root of their individual covariance values (reference 4). The estimate is not worth knowing to a precision that is higher than P_x indicates (reference 5).

The algorithm for the nonlinear least squares estimator routine is in Appendix C.

Bayes Filter Development

The Bayes Filter is basically a sequential nonlinear least squares algorithm. It allows the estimate and covariance of one estimator to serve as data to another estimator. If the first estimate was carefully done, it contains all the information from the previous segment of data worth remembering as well as a covariance matrix to indicate how much the estimate can be trusted (reference 5).

The Bayes Filter forms an estimate from two types of data. The "new" data $\bar{z}$ consists of observation data and the "old" data consists of the previous estimate $\bar{x}(-)$. The observation relationship, as was defined by Equation (3-13), is

$$\bar{z} = G(\bar{x},t) \tag{3-38}$$

and the observation relationship for the previous estimate is

$$\overline{x}(-) = I\overline{x} \qquad (3\text{-}39)$$

while the observation matrix can be written as

$$T = \left[\begin{array}{c} I \\ \hline T_z \end{array}\right] \qquad (3\text{-}40)$$

where T_z = observation matrix for the "new" data.

The covariance matrix representing the "old" estimate and the "new" data is:

$$Q = \left[\begin{array}{c|c} P(-) & 0 \\ \hline 0 & Q_z \end{array}\right] \qquad (3\text{-}41)$$

where

$P(-)$ = covariance of "old" estimate
Q_z = covariance of "new" estimate

and the residual vector is defined as

$$\overline{r} = \left[\begin{array}{c} \overline{x}(-) - \overline{x}_{ref} \\ \hline \overline{z} - \overline{G}(\overline{x}) \end{array}\right] \qquad (3\text{-}42)$$

where the obvious starting point for the reference trajectory is the "old" estimate. The familiar form of least squares is then used to estimate the new state x(+). Recalling Equation 3-37, the inverse covariance matrix for the new data is:

$$P^{-1}(+) = \left(P^{-1}(-),\ T_z^T\ Q_z^{-1} \right) \left[\begin{array}{c} I \\ \hline T_z \end{array}\right]$$

$$= P^{-1} + T_z^T\ Q_z^{-1} T_z \qquad (3\text{-}43)$$

The correction to the state can then be defined as:

$$\delta \bar{x}(t_o) = P(+)\ T^T\ Q^{-1}\ \bar{r}$$

$$= P(+)\ (\ P(-),\ T_z{}^T\ Q_z{}^{-1}\ ) \begin{bmatrix} \bar{x}(-) - \bar{x}_{ref} \\ \hline \bar{r}_z \end{bmatrix}$$

$$= P(+)\ [\ P^{-1}(-)\ \{\bar{x}(-) - \bar{x}_{ref}\}\ +\ T_z{}^T\ Q_z{}^{-1}\ \bar{r}_z\ ] \quad (3\text{-}44)$$

An algorithm for the Bayes Filter is contained in Appendix D.

IV. STAGING ESTIMATION

The estimation routine developed by Capt. Vallado in his thesis (reference 4) was only marginally successful in dealing with a staging event. He achieved partial success when the staging event occurred at the end of a Bayes Filter segment. This problem was compounded when the dynamics were altered to allow for multi-staged vehicles. The staging estimator will be responsible for detecting that an event has taken place, estimating the time of staging, exhaust velocity, and mass ratio of the next stage, and returning to the Bayes Filter to continue processing sensor data.

Staging Event Detection

The change in dynamics can be utilized in determining that a staging event has taken place. After staging has occurred, the Haming numerical integrator is still propagating the old dynamics in time while the sensor data is reflecting a distinct change in acceleration. This "out-of-track" condition can be reflected analytically as follows:

$$\bar{r}_{res,IJK} = \bar{r}_{HAMING} - \bar{r}_{OBSER} \qquad (4\text{-}1)$$

where

$\bar{r}_{res,IJK}$ = difference between position vectors from Haming and from the observations, both in the IJK frame

$\bar{r}_{haming}$ = position vector from Haming reflecting the dynamics of the "old" stage

$\bar{r}_{obser}$ = actual vehicle position as the sensor sees it

In order to compute the "in-track" residual, the sensor ob-

servation, consisting of range, azimuth, and elevation data, must be converted to a position vector in the IJK frame. Recalling the observation relationships developed in Chapter 2 (see Figure 5), the $\bar{\rho}$ vector can be represented in the SEZ frame by:

$$\bar{\rho}_{SEZ} = \rho\cos(az)\cos(el)\,\hat{S} + \rho\sin(az)\cos(el)\,\hat{E} + \rho\sin(el)\,\hat{Z} \qquad (4\text{-}2)$$

where

ρ = $|\bar{\rho}_{SEZ}|$, the sensor range measurement
az = azimuth angle measured from the South (CCW)
el = elevation angle measured up from the S and E plane

Now that we have the $\bar{\rho}$ vector from the sensor site to the observation site, it must now be converted to the IJK frame as follows:

$$\bar{\rho}_{IJK} = \hat{S} \quad \hat{E} \quad \hat{Z} \quad \bar{\rho}_{SEZ} \qquad (4\text{-}3)$$

where the transformation matrix was developed in Chapter 2 (Equation (2-16)). Once we have the $\bar{\rho}$ vector in the IJK frame, the position vector in the IJK frame can be defined as:

$$\bar{r}_{obser} = \bar{r}_s + \bar{\rho}_{IJK} \qquad (4\text{-}4)$$

where

$\bar{r}_{obser}$ = the position vector of the observation point converted to the IJK frame
$\bar{r}_s$ = position vector of the radar site in the IJK frame

From Equation (4-1), it is obvious that once the staging

event has taken place, the deviation between the numerical solution and the observation data will increase with time. It is now left to define at which point in this out-of-track condition to declare that a staging event has taken place. For this, an in-track covariance matrix must be developed using the methods introduced in Chapter 3.

The in-track residual, Equation (4-1), will aid in the formulation of the appropriate covariance value. What is needed is an estimate of the accuracy of an in-track condition as a function of the observation data accuracy. To begin, the in-track residual will be redefined in the direction of the thrust since the deviation in the trajectory is due to the change in thrust. Therefore, Equation (4-1) can be expressed as:

$$\Delta r_{intrk} = \overline{r}_{res,ijk} \cdot \frac{\overline{V}_{ijk}}{\left|\overline{V}_{ijk}\right|} \qquad (4\text{-}5)$$

where

$\overline{r}_{res,ijk}$ = defined in Equation (4-1)
Δr_{intrk} = intrack residual (scalar) in direction of thrust
$\overline{V}_{ijk}$ = velocity vector for vehicle, available from Haming

With the in-track residual now expressed in the direction of the velocity vector, the covariance of the in-track residual can be defined as:

$$\sigma^2_{intrk} = E\,(\delta r^2_{intrk}) \qquad (4\text{-}6)$$

where

σ^2_{intrk} = represents the 1 x 1 in-track covariance matrix

$E(\cdot)$ = expectation operator introduced in Chapter 3

δr_{intrk} = differential of Equation (4-5)

To compute the covariance matrix, the differential of Equation (4-5) must be defined. Allowing that the residual is independent of the frame computed in, Equation (4-5) can be rewritten as follows:

$$\Delta r_{intrk,ijk} = \Delta r_{intrk,sez} = \overline{r}_{res,sez} \cdot \frac{\overline{V}_{sez}}{|\overline{V}_{sez}|} \quad (4\text{-}7)$$

Recalling Equation (2-17), the velocity component can be written as:

$$\frac{\overline{V}_{sez}}{|\overline{V}_{sez}|} = [\hat{S} \mid \hat{E} \mid \hat{Z}]^T \quad \frac{\overline{V}_{ijk}}{|\overline{V}_{ijk}|} \quad (4\text{-}8)$$

Since all the uncertainty is assumed to be contained in the observation data (i.e. assuming the problem is modeled adequately), the differential of Equation (4-5) can be expressed as:

$$\delta \overline{r}_{res,sez} = - \delta \overline{r}_{observation} \quad (4\text{-}9)$$

Using Equation (4-4) and assuming the position vector of the sensor (land based or space based) is known to a much higher degree of accuracy, Equation (4-9) may be written as:

$$\delta \overline{r}_{res,sez} = - \delta \rho_{sez} \quad (4\text{-}10)$$

where $\delta \bar{\rho}_{SEZ}$ = differential of Equation (4-2). With Equation (4-8) and (4-10), Equation (4-7) can be rewritten as:

$$\delta \Delta r_{intrk} = \delta \bar{\rho}_{SEZ} \cdot \left\{ [\, \hat{S} \mid \hat{E} \mid \hat{Z} \,]^T \quad \frac{\bar{V}_{sez}}{|\bar{V}_{sez}|} \right\} \quad (4\text{-}11)$$

The differential of Equation (4-2) in component form is:

$$\begin{aligned} \delta \bar{\rho}_{SEZ} = {} & [\, \cos(az)\cos(el)\,\delta\rho - \rho\sin(az)\cos(el)\,\delta az - \\ & \rho\cos(az)\sin(el)\,\delta el \,]\,\hat{S} + [\, \sin(az)\cos(el)\,\delta\rho \\ & + \rho\cos(az)\cos(el)\,\delta az - \rho\sin(az)\sin(el)\,\delta el \,]\,\hat{E} \\ & + [\, \sin(el)\,\delta\rho + \rho\cos(el)\,\delta el \,]\,\hat{Z} \qquad (4\text{-}12) \end{aligned}$$

Using Equation (4-12) and performing the scalar (dot) product, Equation (4-11) becomes:

$$\begin{aligned} \delta \Delta r_{intrk} = {} & - v^{-1}_{sez} [\, v_S \cos(az)\cos(el) \\ & + v_E \sin(az)\cos(el) + v_Z \sin(el) \,]\,\delta\rho - \\ & v^{-1}_{sez} [v_E\, \rho \cos(az)\cos(el) \\ & - v_S\, \rho\sin(az)\cos(el) \,]\,\delta az - \\ & v^{-1}_{sez} [\, v_Z\, \rho\cos(el) - v_S\, \rho\cos(az)\sin(el) \\ & - v_E\, \rho\sin(az)\sin(el)]\,\delta el \end{aligned}$$

$$(4\text{-}13)$$

where

$$\bar{V}_{sez} = v_S\,\hat{S} + v_E\,\hat{E} + v_Z\,\hat{Z}$$

$$v_{sez} = [\, v_S 2 + v_E 2 + v_Z 2 \,]^{1/2}$$

In most radar systems the uncertainty of the measurements for range, azimuth, and elevation are independent of one another. For this research, it will be assumed that the three measure-

ments are independent of each other and when Equation (4-13) is squared, the cross terms will be neglected. Therefore, Equation (4-6) can be rewritten as:

$$\sigma^2_{intrk} = v^{-2}_{sez} \; [\; v_s \cos(az)\sin(el) + v_e \sin(az)\cos(el) + v_z \sin(el) \;]^2 \; \sigma^2_{range} \; + \; v^{-2}_{sez} \; [\; v_e\, \rho \cos(az)\cos(el) - v_s\, \rho \sin(az)\cos(el) \;]^2 \; \sigma^2_{az} + v^{-2}_{sez} \; [\; v_z\, \rho \cos(el) - v_s\, \rho \cos(az)\sin(el) - v_e\, \rho \sin(az)\sin(el) \;]^2 \; \sigma^2_{el} \tag{4-14}$$

Now that an in-track error value has been derived in terms of the individual errors of the observation data, it is important to establish the criteria for determining whether a staging event, an "out-of-track" condition, exists. This will be done by comparing the in-track residual to the "3-sigma" ($3 * \sigma_{intrk}$) value of the error as derived in Equation (4-14). If the in-track residual is outside the 3-sigma envelope for three successive observation points, a staging event will be declared. Using three sucessive points as the criteria was chosen to preclude the possibility of a single "bad" observation point triggering the staging event detector.

The probability of the value of the in-track residual being within the 3-sigma interval is slightly over 99 percent (reference 5).

Nonlinear Least Squares Staging Estimator

Once the staging event has been detected, the staging

estimator is responsible for determining the next stage's vehicle characteristics. The change in vehicle exhaust velocity and mass ratio (as defined in Equation (2-8)) result in a definite change in vehicle acceleration. Also, since the dynamics reflect the time of staging as in Equation (2-8), the in-track residual (defined in Equation (4-7) as a difference in position) is assumed to be a direct result of the change in acceleration due to staging. Therefore, it can be written:

$$\Delta r_{intrk} = \int\int_{t_{snew}}^{t} \Delta a_{intrk} \, dt \tag{4-15}$$

where

$$\Delta a_{intrk} = a_{old} - a_{new} \tag{4-16}$$

and

a = Equation (2-5)
t_{snew} = new stage time

In order to develop the dynamics to be used in the staging estimator, the integration of Equation (4-15) must be performed. Recalling Equation (2-8), Equation (4-16) may be written as:

$$\Delta a_{intrk} = \frac{V_{eold} \, M_{old}}{1-M_{old}(t-t_{sold})} - \frac{V_{enew} \, M_{new}}{1-M_{new}(t-t_{snew})} \tag{4-17}$$

Noting that the difference $(t-t_{snew})$ is small after the

staging event occurs, the first term is rewritten as:

$$a_{old} = V_{eold}\, M_{old} \quad [1-M_{old}(ts_{new}-ts_{old})] \quad + \quad [-M_{old}(t-ts_{new})]^{-1} \qquad (4\text{-}18)$$

Equation (4-18) is then expanded in binomial form as:

$$a_{old} = V_{eold}\, M_{old} \quad [1-M_{old}(ts_{new}-ts_{old})]^{-1} \quad + \quad [1-M_{old}(ts_{new}-ts_{old})]^{-2}\, [-M_{old}(t-ts_{new})] \quad + \text{H.O.T.}$$

By neglecting higher order terms, Equation (4-18) can be approximated by:

$$a_{old} \approx \frac{V_{eold}\, M_{old}}{[1-M_{old}(ts_{new}-ts_{old})]} - \frac{V_{eold}\, M^2_{old}}{[1-M_{old}(ts_{new}-ts_{old})]^2} \qquad (4\text{-}19)$$

The second term of Equation (4-17) can also be binomially expanded as:

$$a_{new} = V_{enew}\, M_{new} \quad [1+M_{new}(t-ts_{new}) + \text{H.O.T.}\ ]$$

Again neglecting the higher order terms, it may be rewritten as:

$$a_{new} \approx V_{enew}\, M_{new} + V_{enew}\, M^2_{new}\, (t-ts_{new}) \qquad (4\text{-}20)$$

With the two approximations just developed, Equation (4-17) can be redefined as:

$$\Delta a_{intrk} \approx A_o + B_o\, (t-ts_{new}) \qquad (4\text{-}21)$$

where

$$A_o = \frac{V_{e\,old}\, M_{old}}{[1-M_{old}(t_{s\,new}-t_{s\,old})]} - V_{e\,new}\, M_{new} \qquad (4\text{-}22)$$

and

$$B_o = \frac{V_{e\,old}\, M^2_{old}}{[1-M_{old}(t_{s\,new}-t_{s\,old})]^2} - V_{e\,new}\, M^2_{new} \qquad (4\text{-}23)$$

Integrating Equation (4-17), the dynamics for the nonlinear least squares staging estimator becomes:

$$\Delta r_{intrk} \approx \frac{A_o}{2}(t-t_{s\,new})^2 + \frac{B_o}{6}(t-t_{s\,new})^3 \qquad (4\text{-}24)$$

Now that the basic equations for the staging estimator have been developed, the matrix equations, identified in Chapter 3, are left to be developed. The state vector is defined as:

$$\bar{x} = \begin{bmatrix} V_{e\,new} \\ M_{new} \\ t_{s\,new} \end{bmatrix} \qquad (4\text{-}25)$$

where

$V_{e\,new}$ = exhaust velocity of the next stage
M_{new} = mass ratio ($\dot{m}/m_o$) of next stage
$t_{s\,new}$ = time of staging

Since the three values of the state matrix are essentially constants for a particular stage, Equation (3-2) can be expressed as:

$$\dot{\bar{x}} = 0 \qquad (4\text{-}26)$$

Equation (4-26) simply states that the estimate of the in-track residual does not need to be moved through time using a numerical integrator. This simplifies the algorithm extensively. Equation (3-11) can be expressed as:

$$\phi(t,t_o) = 0 \qquad (4\text{-}27)$$

Therefore, applying the initial condition to Equation (3-11), as specified in Equation (3-12), Equation (4-27) becomes:

$$\phi(t_i,t_o) = I \qquad (4\text{-}28)$$

where

I = Identity matrix

For this simplified version of a nonlinear least squares algorithm, the predicted value for the estimator as defined in Equation (3-13) is:

$$z(t_i) = \Delta r_{intrk} \qquad (4\text{-}29)$$

where the right side of the equation is represented by the right hand side of Equation (4-25) evaluated at observation time t_i and the current state vector estimate. Essentially there is no [G] vector as the right hand side of Equation (4-29) is a scalar.

$$z(t_i) = \frac{A_o}{2}(t_i - t_{snew})^2 + \frac{B_o}{6}(t_i - t_{snew})^3 \qquad (4\text{-}30)$$

Now that the predicted value has been defined, it is left

to formulate the [H] matrix and then the [T] matrix in order to have all the components for the estimator. As defined in Equation (3-19), the [H] matrix can be defined as the partial derivative with respect to the state vector of the right hand side of Equation (4-30). Therefore it can be stated:

$$[H] = [\ H_{11} \ \vdots\ H_{12} \ \vdots\ H_{13}\ ] \qquad (4\text{-}31)$$

where

$$H_{11} = -\frac{M_{new}(t-ts_{new})^2}{2} - \frac{M^2_{new}\ (t-ts_{new})^3}{6}$$

$$H_{12} = -\frac{Ve_{new}(t-ts_{new})^2}{2} - \frac{Ve_{new}\ M_{new}\ (t-ts_{new})^3}{3}$$

$$H_{13} = \frac{Ve_{old}\ M^2_{old}\ (t-ts_{new})^2}{2[1-M_{old}(ts_{new}-ts_{old})]^2}\left\{1+\frac{M_{old}\ (t-ts_{new})}{3[1-M_{old}(ts_{new}-ts_{old})]}\right\}$$

$$-\ (t-ts_{new})\ \left[\ A_o + \frac{(t-ts_{new})\ B_o}{2}\ \right]$$

As defined by Equation (3-22), the [T] matrix can now be expressed as:

$$T(t_i) = H(x_o(t_i), t_i)\ \phi\ (t_i, t_o) \qquad (4\text{-}32)$$

and since the state transition matrix is defined as the identity matrix throughout the staging estimator, this simply leaves

$$[T]_i = [\ H_{11} \ \vdots\ H_{12} \ \vdots\ H_{13}\ ]_i \qquad (4\text{-}33)$$

where i denotes values evaluated at the observation time t_i. With the addition of the [T] matrix, the required matrix equations have been developed as applied to the staging estimator. It is then left to follow the procedure as outlined in Chapter 3 for the nonlinear least squares routine. The only difference is the estimate need not be propagated in time to the next observation point as required by the main program.

Reentering Bayes Filter Estimator

Now that the new stage vehicle parameters and staging time have been estimated, the procedure for returning to the Bayes Filter algorithm must be developed.

The first step is to determine the point in the Bayes Filter data segment that corresponds to the observation data point that occurs just prior to the estimated staging time. Having done that, it is important to compute an estimate of the state vector as defined in Equation (3-1). The staging estimator state vector (Equation (4-25)) represents an initial value for the exhaust velocity and mass ratio of the next stage. An estimation of the position and velocity components of the main state vector (Equation (3-1)) need to be specified just prior to the estimated staging time. This can be done by using of the Haming integrator routine as used by the Bayes Filter algorithm. The estimate of the state vector prior to the segment of observation data containing the staging event is propogated through time to the estimated staging

time. This will serve as an initial value for the position and velocity components of Equation (3-1).

Once an estimate for the next stage's state vector at the estimated staging time has been computed, the observation data segment must be reinitialized to include the data immediately following the staging event. This is accomplished by moving the remaining observation data points to the front of the current segment of data being estimated, and then adding more data from the sensor data file to fill the current segment array.

Now it is left to compute a new $P^{-1}(-)$ term (Equation (3-40)) at the new epoch time specified by the estimated staging time. With the P(-) matrix available at the last epoch time and recalling the idea of the state transition matrix, it can be stated:

$$P(t_i) = \phi(t_i, t_o)\, P(t_o)\, \phi^T(t_i, t_o) \qquad (4\text{-}34)$$

where

$P(t_o)$ = covariance matrix at old epoch time
$\phi(t_i, t_o)$ = Equation (3-11), designed into the Haming routine

The inverse of Equation (4-34) is then computed as:

$$P^{-1}(-) = [\,\phi(t_i, t_o)\, P(t_o)\, \phi^T(t_i, t_o)\,]^{-1} \qquad (4\text{-}35)$$

Having accomplished this, the Bayes Filter is then allowed to continue processing data for the next stage.

V. RESULTS AND CONCLUSIONS

The objective of this paper is the detection of a staging event, estimation of the next stage's vehicle parameters (V_e and M) including the staging time, and continuation of the main Bayes Filter algorithm for processing of subsequent stage sensor data. The computer programs presented by Capt. Vallado in his thesis (reference 4) are used extensively in this paper. The various testing routines developed by Capt. Vallado to test the individual portions of the Bayes Filter algorithm will not be repeated in this thesis as it would serve no useful purpose to revalidate an algorithm which has already been successfully used. Bearing this in mind, the first task to accomplish is the verification of the final results attained by Capt. Vallado in his thesis. This will serve as a "check" to verify the programs he was working with at the end of his research. Once this has been accomplished the objectives of this reseach can be investigated.

Truth Model Data Formulation

Early in the program verification phase, it was discovered that although Capt. Vallado's results were exactly duplicated, a problem was encountered. By extending the flight time to generate multi-stage data, the data being generated for stage two reflected a "negative mass" situation. It was discovered that the equations of motion for the launch vehicle did not include the ability to generate

data for more than one stage (Equation (2-6)). Once the dynamics were changed to reflect the multi-staged vehicle (Equation (2-7)), the "negative mass" problem was solved. With confidence that the generated sensor data was correctly being computed, the stage event detection routine was developed.

Staging Event Detection

In Chapter 4, the routine for detection of a staging event was developed. The in-track residual (Equation (4-5)) was computed for each Bayes Filter segment of data during the first iteration of the nonlinear least squares portion of the algorithm. Typical values for in-track residuals computed are listed in Table 2.

Table 2 Typical In-track Residual Values Before Staging

OBSERVATION TIME (TU)	RESIDUAL
0.2008071880069E+00	-0.5166060968207E-13
0.2013029665306E+00	-0.3536990438151E-13
0.2017987450543E+00	-0.6292637105828E-13
0.2022945235780E+00	-0.5920873811415E-13
0.2027903021017E+00	-0.6187166496192E-13

The in-track residual represents the difference in vehicle position that exists between the numerically integrated value (predicted) and the observation data. This difference is computed in the direction of the velocity vector (thrust direction). The "smallness" of the numbers indicates an "in-track" condition exists prior to staging. A typical sequence

of residuals that are encountered after a staging event are listed in Table 3.

Table 3 Typical In-track Residual Values After Staging

OBSERVATION TIME (TU)	RESIDUAL
0.2052691947202E+00	-0.1123816712054E-05
0.2057649732439E+00	-0.2311412768374E-05
0.2072523088151E+00	-0.1266778422883E-04
0.2122100940521E+00	-0.1029856901690E-03
0.2196467719077E+00	-0.4145433471524E-03
Staging ocurred at .2045082E+00	

Noting Table 3, it is apparent that a deviation from the "in-track" condition exists. This is due to the numerical integrator moving the estimate through time using different dynamics than the observation data reflects. Equation (2-8) has changed to reflect a staging event has occurred. A graphical representation of the staging event is presented in Figure 7. Figure 7 represents the in-track residuals plotted as a function of time. The residuals after the staging event increase rapidly.

Now that the staging event is identified graphically, the sharp rise in in-track residual values reflected in Figure 7 can be used to detect the event. In Chapter 4, the procedure for developing an in-track error value (σ_{intrk}, Equation (4-14)) was developed. It is computed as a function of sensor data accuracy as defined in Equation (2-22). Typical values for the in-track error are presented in Table

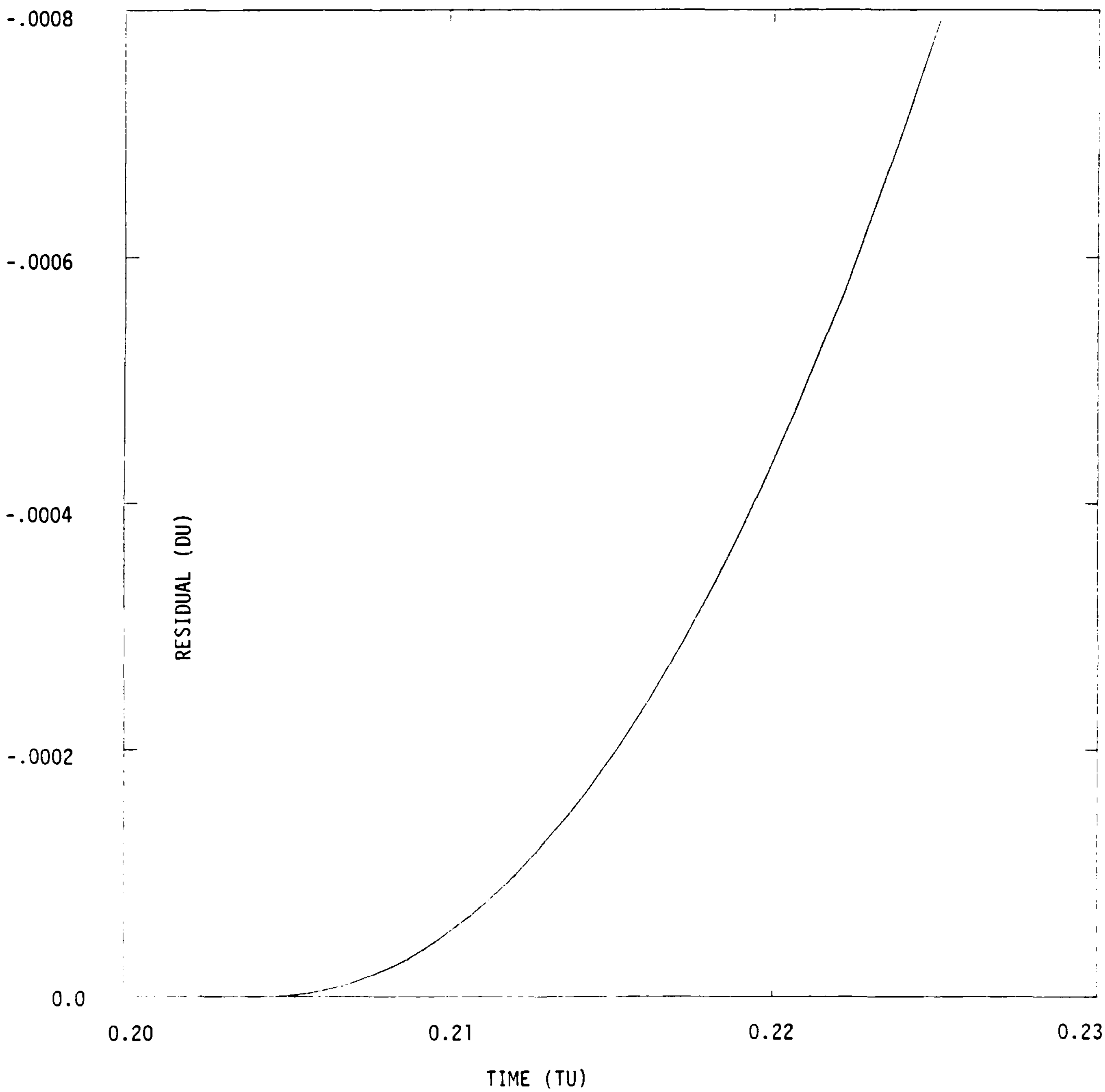

Figure 7 In-track Residual vs Time

4. The value changes very little from segment to segment

Table 4 In-track Error Values

SEGMENT	σ_{intrck}
2	0.1666674317104E-03
4	0.1669833891639E-03
6	0.1674669890607E-03
8	0.1680668838485E-03

as it is just a function of the observation data accuracy in the direction of the velocity vector. This is attributed to the basic assumption that the measurement accuracies do not change with respect to time or range.

With the computed in-track error value (σ_{intrk}) it is a simple process to compare it with the in-track residuals to determine if a staging event has taken place. As defined in Chapter 4, the "3-sigma" value is used to determine a staging event has taken place. When the first three consecutive values of the in-track residual become larger than the "3-sigma" value a staging event is declared. With the detection of the staging event, the next stage's vehicle parameters and staging time must be estimated.

Staging Time and Vehicle Parameter Estimation

The staging estimator developed in Chapter 4 was marginally successful for estimating the staging time and poor for estimating the vehicle parameters for the next stage. Table 5 shows the covariance and state estimation verses the actual values for M, V_e, and t_{stage}. This was accomplished by

Table 5 Covariance and Estimate of Tstage, Ve, and M

COVARIANCE MATRIX AFTER 10 ITERATIONS		
0.5569853112E+04	-0.34634315092E+05	-0.24942132797E+00
0.3463431509E+05	0.21536776000E+06	0.15455896996E+01
0.2494213280E+00	0.15455896996E+01	0.16662358470E-04
ESTIMATION OF STATE VARIABLES AFTER CONVERGENCE		
Ve	M	tstage
-0.213423948E+02	0.15607495538E+03	0.20509324656E+00
ACTUAL STAGE TWO VEHICLE PARAMETERS		
0.3941439263E+00	0.25041207011E+01	0.20450820360E+00

processing all the in-track residual data points for the Bayes Filter segment that the staging event occurred in. Using the convergence criterion defined in Chapter 3, convergence was achieved after one iteration. The data reflects a poor estimation in the Ve and M (a negative Ve is impossible) portion of the state vector and a relatively good one for the staging time. In an attempt to give the estimator more time to compute a better estimate of the state, the criterion for convergence was reduced by two orders of magnitude. The results of this case are reflected in Table 6. Although convergence was not achieved, it was a good example of the inability of the estimator to determine the second stage vehicle parameters.

With confidence that the estimator was functioning as it was designed to, a decision was made to redevelop the staging estimator to allow for only a two-stage estimate.

Table 6 Three-State Estimation

INITIAL ESTIMATE		
Ve	M	tstage
0.39414392630E+00	0.25041207011E+01	0.20450820000E+00
ITERATION # 1 STATE CORRECTIONS		
0.10868680693E+02	-0.67517133822E+02	-0.58288645895E-03
CURRENT STATE ESTIMATE		
0.11262824620E+02	-0.65013013121E+02	0.20392531354E+00
ITERATION # 3 STATE CORRECTIONS		
-0.51235725356E-03	0.15004630567E+04	-0.90516578867E-04
CURRENT STATE ESTIMATE		
0.10053896683E-02	0.14356543640E+04	0.20383418292E+00
ITERATION # 5 STATE CORRECTIONS		
0.43724066445E-02	-0.25322025281E+03	-0.22816475211E-03
CURRENT STATE ESTIMATE		
0.68936211756E-02	-0.17340831041E+02	0.20419900563E+00
ITERATION # 7 STATE CORRECTIONS		
-0.99088477821E-01	-0.40857299492E+01	-0.49256693124E-05
CURRENT STATE ESTIMATE		
0.23073217627E-02	0.47975652083E+03	0.20394230354E+00
ITERATION # 9 STATE CORRECTIONS		
0.32538670180E-02	-0.18809488825E+03	-0.21591223996E-03
CURRENT STATE ESTIMATE		
0.65521596420E-02	0.53432099068E+02	0.20419681910E+00
EXACT VALUES		
0.39414392630E+00	0.25041207011E+01	0.20450820000E+00

Staging Estimator for Two-State System

Recalling the problems encountered trying to estimate the exhaust velocity and mass ratio separately (first indication of possible observability problem), it was decided to alter the staging estimator and attempt to estimate the product of the two vehicle parameters. Therefore, Equation (4-25) can be rewritten as:

$$\overline{x} = \begin{bmatrix} MVe_{NEW} \\ ts_{NEW} \end{bmatrix} \tag{5-1}$$

It was also decided that the cubic term of Equation (4-24) be dropped. This term is only of significance far away from the staging event (where $t - t_{stage}$ is no longer small) and of maximum concern just prior to a staging event. As this algorithm is designed to process a small number of data points at a time (Bayes Filter) in order to aid in detection of a staging event, the cubic term would not have a significant amount of time to influence the estimation. Therefore, Equation (4-24) can be rewritten as:

$$\Delta r_{intrk} = \frac{A_o}{2} (t - ts_{new})^2 \tag{5-2}$$

Equation (4-33) must then be changed to:

$$[T]_i = [\ H_1 \ \vdots \ H_2\ ]_i \tag{5-3}$$

where

$$H_1 = -\frac{1}{2}(t - tsnew)^2 \tag{5-4}$$

$$H_2 = \frac{Ve_{old}\ M^2_{old}}{2[1-M_{old}(ts_{new}-ts_{old}]^2}\ (t-ts_{new})^2 - A_o\ (t-ts_{new}) \qquad (5-5)$$

Once the conversion of the necessary matrices is accomplished, the algorithm is basically identical to the one developed for the three-state system.

After making the necessary changes to the staging algorithm, the same test case was performed with much better results. Table 7 lists the covariance, last estimate of the state at convergence, and the actual values.

Table 7 Covariance and Estimate of VeM and T_{stage}

COVARIANCE MATRIX AT TIME OF CONVERGENCE	
0.82116001614E+00	-0.18430617242E-02
-0.18430617242E-02	0.45527066070E-05
ESTIMATION OF STATE AT CONVERGENCE	
VeM	t_{stage}
0.59200295714E+00	0.20473674506E+00
ACTUAL VALUES FOR STAGE TWO	
0.986983965E+00	0.2045082036E+00

Table 7 represents the output of the staging estimator once it has achieved convergence. The covariance indicates that the value of VeM is not known as well as the staging time but the values do reflect a much better estimate than the three-state system achieved. With the results of the two-state system, the next logical step was to pass the estimate

of the next stage parameters back to the Bayes Filter to continue processing sensor data.

Reentering the Bayes Filter Algorithm

The procedure for reentering the Bayes Filter outlined in Chapter 4 was accomplished. Table 8 represents the typical problems encounterd by the Bayes Filter in estimating the state vector (Equation (3-1)) components involving the launch vehicle parameters (V_e and M) of the next stage. The

Table 8 Convergence of Main Filter After Staging

COVARIANCE OF V_e AND M	
0.1305007E+00	-0.1032639E+00
-0.1032639E+00	0.1348049E+00
STATE ESTIMATE AT CONVERGENCE	
V_e	M
0.7725107E+00	0.1300567E+01
EXACT VALUES	
0.3941439E+00	0.2504121E+01

estimated values of exhaust velocity and mass ratio passed back to the main loop by the staging estimator represents an initial value for estimation of the state vector (Equation (3-1)) for the next stage. The initial value of exhaust velocity and mass ratio was chosen by assuming the exhaust velocity to be the same as the previous stage and dividing the estimated product by the exhaust velocity to compute the initial

value for the mass ratio. The problem encountered seems to indicate that the Bayes Filter had little idea as to the values of stage two exhaust velocity and mass ratio. With this in mind, it was left to devise an alternative means to reenter the Bayes Filter.

Noting that the staging estimator, after convergence, represents an estimate of the product VeM with an associated covariance matrix, it was decided to take advantage of the errors associated with the staging estimator. Since the covariance matrix is an indication as to how much the state vector can be "trusted" (reference 5), a method was devised to alter the covariance matrix of the main Bayes Filter. This will indicate to main Bayes Filter that the values of Ve and M of the next stage are no longer as reliable as the first stage values were. Representing the estimate of the mass ratio to the Bayes Filter as

$$M_{new} = \frac{[\ Ve\ M\ ]_{new}}{V_{old}} \qquad (5\text{-}6)$$

and recalling the procedure for finding the error associated with multiple estimates (Chapter 4), the differential of Equation (5-6) is computed as:

$$\delta(M_{new}) = \frac{\delta\,(Ve\ M)_{new}}{Ve_{old}} - \frac{(Ve\ M)_{new}}{(Ve_{old})^2}\,\delta\,(Ve_{old}) \qquad (5\text{-}7)$$

Squaring both sides of Equation (5-7) yields:

$$[\,\delta(M_{new})]^2 = \frac{[\,\delta(VeM)_{new}]^2}{(Ve_{old})^2} + \frac{(VeM)^2_{new}}{(Ve_{old})^4}[\,\delta(Ve_{old})]^2 - \frac{2\,(VeM)_{new}\,[\,\delta(Ve_{old})]\,[\,\delta(VeM)_{new}]^2}{(Ve_{old}{}^3} \tag{5-8}$$

Applying the expectation operator (E) and neglecting cross terms leaves:

$$\sigma^2{}_M = Ve^{-2}_{old}\left[\,\sigma^2{}_{vem\ new} + \frac{(VeM)^2_{new}}{(Ve_{old})^2}\,\sigma^2{}_{Veold}\,\right] \tag{5-9}$$

Equation (5-9) represents the error associated with the estimate of the mass ratio to be passed back to the Bayes Filter to start stage two data processing. The error associated with the estimate of the exhaust velocity is estimated as twice the existing value. It was also decided that the Bayes Filter segment after the staging event be allowed to process more data points to estimate the state better. Appendix F represents the results of the procedure just discussed. It represents a much better estimate of the state vector (Equation (3-1)) for the second stage.

The Observability Problem

The many procedures and cases that were attempted seem to all have the same basic problem. In all instances, the last two components of the state vector (Ve and M), have caused problems for the estimator. Capt. Gross, in his paper

(reference 2), stated that a better approach may be to estimate the product V_eM since it appears as a product in all instances, and also estimate the mass ratio since it does appear in the denominator of Equation (2-8). Although this change may be of some help to the main portion of the Bayes Filter, it would not seem to help the staging estimator since the next stage mass ratio does not appear in the dynamics alone. The cubic term that appears in Equation (4-24), although it does contain the multiple product term ($V_{eNEW}\ M_{NEW}^2$), is not utilized. Since the data available to estimate the staging state vector (in-track residuals) is a relatively small segment of the sensor data, the cubic term is not evaluated for a long enough time period to be of any use. This is due to the fact that the main algorithm was designed to process small segments of data (Bayes Filter) in order to isolate the staging event for detection. Capt. Vallado, in his paper (reference 4), encountered the same problem to some degree. The Bayes Filter had a difficult time estimating the state to convergence unless the initial value contained some degree of accuracy. In most cases, the launch point of the vehicle being known, the position and velocity components of of the state vector can be estimated with the proper degree of accuracy. As the vehicle is, in most cases, of an unknown type, the exhaust velocity and the mass ratio will not be estimated with the same degree of accuracy. If the initial value deviated from the exact value

in the fourth or fifth decimal place, the last two components of the state vector would try to drastically correct itself and the estimator would fail. As seen in Appendix F, the portion of main filter covariance corresponding to the mass ratio and exhaust velocity are several orders of magnitude larger than the errors associated with the velocity and position components. This would tend to support the observability problem theory.

A further case was considered using noisy data. The data was generated using the methods as outlined in Chapter 2. While the estimator was attempting to converge on the noisy data, it failed after two iterations. The last two estimates of the state would drastically change and the estimator would fail. This problem persisted even when the "random" noise added to the observation data was decreased. Upon starting the estimator with the exact values, it still failed after only one more iteration.

Conclusion

In this paper, the estimation of ICBM performance parameters was investigated. Specifically, the detection of a staging event, estimation of next stage vehicle parameters including staging time, and subsequent estimation of next stage performance parameters. While the detection of the staging event was successful, it was quite apparent that a problem existed determining the exhaust velocity (V_e) and mass ratio (M) independently. Much more success was achieved

by estimating the product of the two parameters. This problem was also encountered in the main Bayes Filter developed by Capt. Vallado (reference 4). If the initial estimate of the state was not of at least fourth place accuracy, the estimator failed.

Recommendations for further study would be to redefine the main state vector to include the product of exhaust velocity and mass ratio instead of the present configuration. This would involve the redesign of the Bayes Filter developed by Capt. Vallado (reference 4). Once this has been accomplished, methods could be developed to make use of the estimated product of exhaust velocity and mass ratio to gain any launch vehicle parameter information.

BIBLIOGRAPHY

1. Bate, Roger R., Mueller, Donald D., and White, Jerry E. Fundamentals of Astrodynamics. New York : Dover Publications, Inc., 1971

2. Gross, Donald W., "Estimation of Launch Vehicles Performance Parameters from Two Orbiting Sensors", Masters Thesis. Wright Patterson AFB, Ohio : Air Force Institute of Technology, December 1982

3. "Titan III Propulsion Systems Handbook", Integrated Logistics Support, Aerojet Liquid Rocket Company, Sacramento, July 1983

4. Vallado, David A., "Estimation of Launch Vehicle Performance Parameters", Masters Thesis. Wright Patterson AFB, Ohio : Air Force Institute of Technology, December 1984

5. Wiesel, William E., Jr. Lecture material distributed in MC731, Modern Methods of Orbit Determination. School of Engineering, Air Force Institute of Technology, Wright Patterson AFB, Ohio, 1985

6. Wiesel, William E., Jr. Lecture material distributed in MC533, Problems in Spaceflight. School of Engineering, Air Force Institute of Technology, Wright Patterson AFB, Ohio, 1985

7. Wiesel, William E., Jr. Lecture material destributed in MC636, Advanced Astrodynamics. School of Engineering, Air Force Institute of Technology. Wright Patterson AFB, Ohio 1985

APPENDIX A

TRUTH MODEL PROGRAMS

Description

The purpose of this program is to provide the simulated sensor data needed for checkout of the Bayes Filter estimator as listed in Appendix E. It accomplishes this by numerically integrating the equations of motion as developed in Chapter 2. A brief description of the subroutines for this program are as follows:

LSTIME

Calculates the local sidereal time for the sensor sites.

RADST

Calculates the position vector of the sensor site. If a land based sensor is chosen, the user is asked to input the longitude and latitude in degrees and the elevation in DU's. If a space based sensor is chosen, the user is asked to input the orbital elements as described in Capter 2.

RANDV

Calculates the position and velocity vectors given the initial orbit data. The subroutine uses the eccentric anomoly calculation and the Newton Raphson method to find the mean anomoly.

MAG

Computes the magnitude of a vector.

CROSS

Computes the cross product of two vectors.

Haming

This subroutine is a fourth order differential equations integrator. It carries four copies of the state vector along. It then extrapolates them to find the next value. It then corrects the answer to find the new value of the state vector. A more detailed description is contained in Capt. Vallado's thesis (reference 4).

RHS

Calculates the equations of motion and equation of variation for the problem being evaluated. It is only used as a data source for the Haming subroutine.

VEHD

Contains the algorithm to simulate the staging of an launch vehicle.

RAZEL

Computes the sensor data (ρ, az, el) from the $\bar{r}$ vector as computed by the numerical integrator. Also contains the algorithm to compute the transformation matrices as defined in Chapter 2.

```
      PROGRAM TRUTH

C     THE COMMON TERMS

      COMMON /HAM/ T,Y(72,4),F(72,4),ERR(72),N,DT,MODE,TEPOCH
      DOUBLE PRECISION T,Y,F,ERR,DT,TEPOCH
      INTEGER N,NXT,MODE

C     ALL OTHER VARIABLES

      INTEGER NIT,INA,INB,INCC,IND,INE,INF,INO
      DOUBLE PRECISION SIGAZ,LP,LLAT,LLON,LRS(0:3),IVEL,
     +   LLST,RAD,TP,RHO,AZ,EL,TO,DSEED,SIGRHO,LVV(0:3),NUDGE,
     +   R(0:3),V(0:3),RS(0:3),RCV(0:3),TRM(3,3),SIGEL,
     +   RM(0:3),VM(0:3),DOT,GAMMA,TM
      CHARACTER TYPE,ANS,TYPED

C     BEGIN THE PROGRAM

2     FORMAT(6X,'R (KM)',8X,'V (FT/SEC)',6X,'GAMMA (DEG)',6X,
     +   'TIME (SEC)',4X,'VE (DU/TU)',4X,'M   ')
4     FORMAT(2X,4E20.13,/,2X,4E20.13,/,2X,'THE INITIAL TIME IS ',
     +   F6.4)
6     FORMAT(2(1X,F14.6),4(2X,F14.10))
12    FORMAT(9X,'X',12X,'Y',12X,'Z',10X,'XDOT',9X,'YDOT',9X,
     +   'ZDOT',10X,'VE',10X,'M')

      OPEN (UNIT=17,FILE='PRODUCT',ACCESS='SEQUENTIAL',STATUS='NEW')
      OPEN (UNIT=14,FILE='TDATA',ACCESS='SEQUENTIAL',STATUS='NEW')

      RAD=3.14159265359D +00/180.0D +00
      NXT=1
      MODE=0
      N=8

C     INPUT INITIAL DATA

      PRINT*,'INPUT THE LENGTH OF THE FLIGHT IN SECONDS, AND TIME'
      READ*,TP,TEPOCH
      T=TEPOCH

      PRINT*,'INPUT STATE VECTOR, OR HAVE IT CALCULATED,Y OR N'
      READ*,TYPED

      IF (TYPED.EQ.'Y') GOTO 5
      IF (TYPED.EQ.'N') GOTO 7

5     PRINT*,'INPUT THE INITIAL STATE VECTOR FOR THE VEHICLE'
      READ*,Y(1,NXT),Y(2,NXT),Y(3,NXT),Y(4,NXT),Y(5,NXT),Y(6,NXT),
     +   Y(7,NXT),Y(8,NXT)

7     IF (TYPED.EQ.'N') THEN

         PRINT*,'LAUNCH SITE POINTS ARE AS FOLLOWS:'
```

```
      PRINT*,'0 PETROPAVLOVSK  53.7N   158.2E'
      PRINT*,'1 VLAIDIVOSTOK   43.5N   132.0E'
      PRINT*,'2 TURYANTUNUM                  '
      PRINT*,'3 PLESEK                       '

      PRINT*,'INPUT THE LAUNCH POINT NUMBER'
      READ*,LP
      IF (LP.EQ.0) THEN
         LLAT=53.7D+00*RAD
         LLON=158.2D+00*RAD
      END IF
      IF (LP.EQ.1) THEN
         LLAT=43.5D+00*RAD
         LLON=132.0D+00*RAD
      END IF
      IF (LP.EQ.2) THEN
         LLAT=1.0D+00*RAD
         LLON=1.0D+00*RAD
      END IF
      IF (LP.EQ.3) THEN
         LLAT=1.0D+00*RAD
         LLON=1.0D+00*RAD
      END IF
      LRS(0)=1.0D+00
      CALL LSTIME(LLST,T,TO,LLON)
      ANS='L'
      CALL RADST(LRS,LLAT,LLST,T,TO,ANS,INO)
      INO=0
      PRINT*,'INPUT INITIAL VELOCITY IN FT/S'
      READ*,IVEL
      IVEL=IVEL/25936.24764D+00
      DO 40 INR=1,3
40       LVV(INR)=IVEL*LRS(INR)
      PRINT*,'HOW MUCH DO YOU WANT TO NUDGE THE VELOCITY'
      READ*,NUDGE
      LVV(3)=LVV(3)+LVV(3)*NUDGE
      DO 44 INW=1,3
         Y(INW,NXT)=LRS(INW)
         Y(INW+3,NXT)=LVV(INW)
44    CONTINUE

   END IF

   TP=TP/806.8118744D+00
   PRINT*,'INPUT THE NUMBER OF ITERATIONS'
   READ*,NIT
   DT=TP/NIT
   NXT=0

C  WRITE INITIAL HEADER DATA

   WRITE(17,*) 'THE INITIAL STATE VECTOR FOR THE MISSILE IS'
   WRITE(17,12)

   INB=0
```

```
      DSEED=88888.D+00
      SIGRHO=1.0D-8
      SIGAZ=1.0D-04
      SIGEL=1.0D-04
      TYPE='N'

C     INITIALIZE HAMING AND RESET THE TIME

      NXT=0
      CALL HAMING(NXT)
      T=TEPOCH
      IF (NXT.EQ.0) STOP
      WRITE(17,4) (Y(INF,NXT),INF=1,8),T
      PRINT*,Y(1,NXT),Y(2,NXT),Y(3,NXT),Y(4,NXT),Y(5,NXT),Y(6,NXT),
     +   Y(7,NXT),Y(8,NXT)
      PRINT*,'Y=',Y(1,NXT)*6378.145D+00,Y(2,NXT)*6378.145D+00,
     +   Y(3,NXT)*6378.145D+00,Y(4,NXT)*7.90536828D+00
      PRINT*,'Y=',Y(5,NXT)*7.90536828D+00,Y(6,NXT)*7.9053628D+00,
     +   Y(7,NXT),Y(8,NXT)

      WRITE(17,2)

C     BEGIN LOOP TO INTEGRATE

      DO 10 INCC=0,NIT

         CALL HAMING(NXT)
         INB=INB+1
         DO 30 IND=1,3
            R(IND)=Y(IND,NXT)
            V(IND)=Y(IND+3,NXT)
30       CONTINUE
         CALL RAZEL(R,V,RHO,AZ,EL,TO,T,RS,TRM,INO)
         IF (IOH.NE.10) THEN
            PRINT*,'DO YOU WANT NOISY DATA, Y OR N'
            READ*,TYPE
            IF (TYPE.EQ.'Y') THEN
               PRINT*,'INPUT A SEED NUMBER FROM 1-21483647D+00'
               READ*,DSEED
            END IF
            IOH=10
         END IF
         IF (TYPE.EQ.'Y') THEN
            RHO=RHO+GGNQF(DSEED)*SIGRHO
            AZ=AZ+GGNQF(DSEED)*SIGAZ
            EL=EL+GGNQF(DSEED)*SIGEL
         END IF
         WRITE(14,*) RHO,AZ,EL,T

         IF (INB.EQ.10) THEN
            DO 34 INS=1,3
               RM(INS)=R(INS)*6378.145D+00
               VM(INS)=V(INS)*25936.2647D+00
34          CONTINUE
            CALL MAG(R)
```

```
              CALL MAG(RM)
              CALL MAG(VM)
              CALL MAG(V)
              DOT=R(1)*V(1)+R(2)*V(2)+R(3)*V(3)
              GAMMA=DACOS(DOT/(R(0)*V(0)))
              TM=T*806.8118744D+00
              WRITE(17,6) RM(0),VM(0),GAMMA/RAD,TM,Y(7,NXT),Y(8,NXT)
              INB=0
           END IF

           IF (INCC.EQ.NIT-1) THEN

C       PRINT OUT FINAL DATA

              PRINT*,'THE FINAL VALUES FOR R AND V ARE:'
              PRINT*,'Y=',Y(1,NXT)*6378.145D+00,Y(2,NXT)*6378.145D+00,
     +           Y(3,NXT)*6378.145D+00,Y(4,NXT)*7.90536828D+00
              PRINT*,'Y=',Y(5,NXT)*7.90536828D+00,Y(6,NXT)*7.90536828D+00,
     +           Y(7,NXT),Y(8,NXT)
              WRITE(17,12)
              WRITE(17,4) (Y(INF,NXT),INF=1,8),T
              WRITE(17,*)
           END IF

10      CONTINUE

        ENDFILE(UNIT=17)
        ENDFILE(UNIT=14)

        END

        SUBROUTINE LSTIME(LST,T,TO,LON)

        DOUBLE PRECISION LST,T,TO,LON

        DOUBLE PRECISION THTGO,TWOPI,GST

        TO=0.0D+00
        TWOPI=6.28318530718D+00
        THTGO=98.85481D+00*(3.14159265359D+00/180.00D+00)
        GST=THTGO+1.0027379093D+00*((T*13.44686457D+00/
     +     1440.0D+00)-TO)
        GST=DMOD(GST,TWOPI)
        LST=GST+LON
        LST=DMOD(LST,TWOPI)

        RETURN
        END

        SUBROUTINE RADST(RS,LAT,LST,T,TO,ANS,INO)

        DOUBLE PRECISION RS(0:3),LAT,LST,T,TO

        CHARACTER ANS
```

```
      DOUBLE PRECISION STA,STE,STI,STOMGA,STARGP,STV(0:3),STM,STN,
     +   STNUO

      INTEGER INO

C     LAND BASED SENSOR

      IF (ANS.EQ.'L') THEN
         IF (INO.EQ.0) THEN
            PRINT*,'INPUT THE ELEVATION OF THE SITE'
            READ*,RS(0)
            INO=10
         END IF
         RS(1)=RS(0)*DCOS(LAT)*DCOS(LST)
         RS(2)=RS(0)*DCOS(LAT)*DSIN(LST)
         RS(3)=RS(0)*DSIN(LAT)

         RETURN

      END IF

C     SPACE BASED SENSOR

      IF (ANS.EQ.'S') THEN
         IF (INO.EQ.0) THEN
            PRINT*,'INPUT THE TRACKING SAT ORBIT DATA, A, E, I, OMEGA, ARGP'
            READ*,STA,STE,STI,STOMGA,STARGP
         END IF
         STN=DSQRT(1/(STA*STA*STA))
         STM=STN*(T-T0)
         CALL RANDV(STA,STE,STI,STOMGA,STARGP,STNUO,STM,RS,STV)
         CALL MAG(RS)
         IF (INO.EQ.0) THEN
64          FORMAT(3X,'A',6X,'E',6X,'I',5X,'OMEGA',3X,'ARGP',4X,'M')
66          FORMAT(6(1X,F6.3))
            WRITE(17,*)
            WRITE(17,*) 'THE TRACKING SATELLITE DATA IS'
            WRITE(17,64)
            WRITE(17,66) STA,STE,STI,STOMGA,STARGP,STM
            INO=10
         END IF

      END IF

      RETURN
      END

      SUBROUTINE RANDV(A,E,INC,OMEGA,ARGP,NUO,M,R,V)

      DOUBLE PRECISION A,E,INC,OMEGA,ARGP,NUO,M,R(0:3),V(0:3)

      DOUBLE PRECISION RAD,P,EL,EO,MO

      RAD=3.14159265359D+00/180.00D+00
      MO=M*RAD
```

```
      INC=INC*RAD
      ARGP=ARGP*RAD
      OMEGA=OMEGA*RAD
      P=A*(1-E*E)

C     NEWTON RHAPSON ITERATION

      EL=MO
8     EO=EL
      EL=EO-(EO-E*DSIN(EO)-MO)/(1.0D+00-E*DCOS(EO))
      IF (DABS(EL-EO).GT.1.0D-12) THEN
         EL=EO-(EO-E*DSIN(EO)-MO)/(1.0D+00-E*DCOS(EO))
         PRINT*,'EL=',EL
         GOTO 8
      END IF

C     FIND THE VALUE OF THE TRUE ANOMALY

      NUO=DATAN2((DSQRT(1.0D+00-E*E))*DSIN(EL)/(1.0D+00-E*
     +   DCOS(EL)),(E-DCOS(EL))/(E*DCOS(EL)-1.0D+00))

C     POSITION AND VELOCITY VECTORS

      R(1)=P*DCOS(NUO)/(1.0D+00+E*DCOS(NUO))
      R(2)=R(1)*DTAN(NUO)
      R(3)=0.0D+00
      V(1)=-DSIN(NUO)/DSQRT(P)
      V(2)=(E+DCOS(NUO))/DSQRT(P)
      V(3)=0.0D+00

      RETURN
      END

      SUBROUTINE MAG(RX)

      DOUBLE PRECISION RX(0:3)

      RX(0)=DSQRT(RX(1)*RX(1)+RX(2)*RX(2)+RX(3)*RX(3))

      RETURN
      END

      SUBROUTINE CROSS(RIN,VIN,VX)

      DOUBLE PRECISION RIN(0:3),VIN(0:3),VX(0:3)

      VX(1)=RIN(2)*VIN(3)-VIN(2)*RIN(3)
      VX(2)=-RIN(1)*VIN(3)+VIN(1)*RIN(3)
      VX(3)=RIN(1)*VIN(2)-VIN(1)*RIN(2)
      CALL MAG(VX)

      RETURN
      END

      SUBROUTINE HAMING(NXT)
```

```
      COMMON /HAM/T,Y(72,4),F(72,4),ERREST(72),N,DT,MODE,TEPOCH

      DOUBLE PRECISION T,Y,F,ERREST,DT,TEPOCH

      INTEGER N,MODE,NXT

      INTEGER IDA,IDB,IDC,IDD,IDE,IDF,IDG,IDH,IDI,IDJ,IDK,IDL,IDM,IDN

      DOUBLE PRECISION TOL,HH,XO

C     THE VARIABLES ARE USED AS FOLLOWS
C     T                INDEPENDENT VARIABLE (TIME)
C     Y(72,4)          STATE VECTOR IN 4 COPIES
C     F(72,4)          EQUATIONS OF MOTION, 4 COPIES
C                         CALL RHS(NXT) UPDATES ENTRY IN F
C     ERREST           ESTIMATE OF TRUNCATION ERROR
C     N                NUMBER OF EQUATIONS BEING INTEGRATED
C     DT               TIME STEP
C     MODE             0 FOR EOM.  1 FOR EOM AND EOV

      TOL=1.0D-12
      IF (NXT) 190,10,200

C     SWITCH ON STARTING ALGORITHM OR NORMAL PROPOGATION
C     THIS IS HAMINGS STARTING ALGORITHM...A PREDICTOR-CORRECTOR
C     NEEDS 4 VALUES OF THE STATE VECTOR , AND YOU ONLY HAVE 1, THE
C     I.C.  HAMING USES PRICARD ITERATION (SLOW AND PAINFULL) TO GET
C     THE OTHER THREE.
C     IF IT FAILS, NXT= 0 ON EXIT, OTHERWISE, NXT=1, AND IT'S OK.

10    XO=T
      HH=DT/2.0D+00
      CALL RHS(1)
      DO 40 IDA=2,4
         T=T+HH
         DO 20 IDB=1,N
            Y(IDB,IDA)=Y(IDB,IDA-1)+HH*F(IDB,IDA-1)
20       CONTINUE
         CALL RHS(IDA)
         T=T+HH
         DO 30 IDC=1,N
            Y(IDC,IDA)=Y(IDC,IDA-1)+DT*F(IDC,IDA)
30       CONTINUE
         CALL RHS(IDA)
40    CONTINUE
      IDD=-10
50    IDE=1
      DO 120 IDF=1,N
         HH=Y(IDF,1)+DT*(9.0D+00*F(IDF,1)+19.0D+00*F(IDF,2)-
     +     5.0D+00*F(IDF,3)+F(IDF,4))/24.0D+00
         IF (DABS(HH-Y(IDF,2)).LT.TOL) GOTO 70
         IDE=0
70       Y(IDF,2)=HH
         HH=Y(IDF,1)+DT*(F(IDF,1)+4.0D+00*F(IDF,2)+F(IDF,3))/3.0D+00
```

```
              IF (DABS(HH-Y(IDF,3)).LT.TOL) GOTO 90
              IDE=0
 90           Y(IDF,3)=HH
              HH=Y(IDF,1)+DT*(3.0D+00*F(IDF,1)+9.0D+00*F(IDF,2)+
     +           9.0D+00*F(IDF,3)+3.0D+00*F(IDF,4))/8.0D+00
              IF (DABS(HH-Y(IDF,4)).LT.TOL) GOTO 110
              IDE=0
 110          Y(IDF,4)=HH
 120       CONTINUE
           T=X0
           DO 130 IDG=2,4
              T=T+DT
              CALL RHS(IDG)
 130       CONTINUE
           IF (IDE) 140,140,150
 140       IDD=IDD+1
           IF (IDD) 50,280,280
 150       T=X0
           IDE=1
           IDD=1
           DO 160 IDH=1,N
              ERREST(IDH)=0.0
 160       CONTINUE
           NXT=1
           GOTO 280
 190       IDD=2
           NXT=IABS(NXT)

 C         THIS IS HAMINGS NORMAL PROPAGATION LOOP

 200       T=T+DT
           IDL=MOD(NXT,4)+1
           GOTO (210,230),IDE

 C         PERMUTE THE INDEX NXT MODULO 4

 210       GOTO (270,270,270,220),NXT
 220       IDE=2
 230       IDI=MOD(IDL,4)+1
           IDJ=MOD(IDI,4)+1
           IDK=MOD(IDJ,4)+1

 C         THIS IS THE PREDICTOR PART

           DO 240 IDM=1,N
              F(IDM,IDI)=Y(IDM,IDL)+4.0D+00*DT*(2.0D+00*F(IDM,IDK)-
     +           F(IDM,IDJ)+2.0D+00*F(IDM,IDI))/3.0D+00
              Y(IDM,IDL)=F(IDM,IDI)-0.925619835D+00*ERREST(IDM)
 240       CONTINUE

 C         NOW THE CORRECTOR - FIX UP THE EXTRAPOLATED STATE
 C         BASED ON THE BETTER VALUE OF THE EQUATIONS OF MOTION

           CALL RHS(IDL)
           DO 250 IDN=1,N
```

```
      Y(IDN,IDL)=(9.0D+00*Y(IDN,IDK)-Y(IDN,IDI)+3.0D+00*DT*
     +   (F(IDN,IDL)+2.0D+00*F(IDN,IDK)-F(IDN,IDJ)))/8.0D+00
      ERREST(IDN)=F(IDN,IDI)-Y(IDN,IDL)
      Y(IDN,IDL)=Y(IDN,IDL)+0.0743801653D+00*ERREST(IDN)
250   CONTINUE
      GOTO (260,270),IDD
260   CALL RHS(IDL)
270   NXT=IDL

280   RETURN
      END

      SUBROUTINE RHS(NXT)

      COMMON /HAM/T,Y(72,4),F(72,4),ERR(72),N,DT,MODE,TEPOCH

      DOUBLE PRECISION T,Y,F,ERR,DT,TEPOCH

      INTEGER N,MODE,NXT

      INTEGER IRA,IRB,IRC,IRD,IRE,IRF,IRG,IRH,IRI,IRJ,IRK

      DOUBLE PRECISION R32,V32,VEL,VAT,VVE,VEM,MASS,ACC,R52,AM(8,8),
     +   MASSO,MDOT,VE,TSTAGE

C     THIS DATA STATEMENT HARDWIRES THE PARTS OF THE
C     A MATRIX WHICH ARE NEVER CHANGED ...ONLY THE MIDDLE
C     3 ROWS CHANGE EACH TIME

      DO 10 IRA=1,8
         DO 10 IRB=1,3
            AM(IRB,IRA)=0.0D+00
10    CONTINUE
      DO 20 IRC=1,8
         DO 20 IRD=7,8
            AM(IRD,IRC)=0.0D+00
20    CONTINUE
      AM(1,4)=1.0D+00
      AM(2,5)=1.0D+00
      AM(3,6)=1.0D+00

C     THE BASIC FUNCTION OF RHS IS TO CALCULATE THE EQUATIONS
C     OF MOTION (THE F ENTRIES) FROM THE GIVEN CURRENT STATE
C     (STORED IN Y) AND THE TIME T

C     EVALUATION OF THE EQUATIONS OF MOTION

C     REFERENCE BATES MEULLER & WHITE, PG 10, N BODY PROBLEM
C     WITH ORIGIN IN SUN.

C     POSITION DOT = VELOCITY VECTOR

      F(1,NXT)=Y(4,NXT)
      F(2,NXT)=Y(5,NXT)
      F(3,NXT)=Y(6,NXT)
```

```
C       VELOCITY DOT = GRAVITY ACCELERATION

        R32=(Y(1,NXT)*Y(1,NXT)+Y(2,NXT)*Y(2,NXT)+
     +     Y(3,NXT)*Y(3,NXT))**1.5D+00
        VEL=(Y(4,NXT)*Y(4,NXT)+Y(5,NXT)*Y(5,NXT)+
     +     Y(6,NXT)*Y(6,NXT))**.5D+00
        V32=VEL**3.0D+00

C       SET THE CONSTANTS WHICH WILL BE USED IN THE A MATRIX

        XMU=1.0D+00
        IF (IRE.EQ.0) THEN
           PRINT*,'INPUT THE TYPE OF MISSILE TO BE EVALUATED'
           PRINT*,'THE CHOICES ARE AS FOLLOWS'
           PRINT*,'SATELLITE IN ORBIT        0'
           PRINT*,'TITAN-IIID                1'
           READ*,IRF
           IRE=10
        END IF
        IF (IRF.EQ.0) THEN
           VVE=1.0D+00
           VEM=1.0D+00
           ACC=0.0D+00
           MASS=1.0D+00
           GOTO 6
        END IF
        CALL VEHD(VE,MDOT,MASSO,T,IRF,TEPOCH,TSTAGE)
        MASS=MDOT/MASSO
        VVE=VEL*VE
        VEM=VE*MASS
        ACC=VE*MASS/(1.0D+00-MASS*(T-TSTAGE))
        Y(7,NXT)=VE
        Y(8,NXT)=MASS

6       F(4,NXT)=-XMU*Y(1,NXT)/R32+ACC*Y(4,NXT)/VEL
        F(5,NXT)=-XMU*Y(2,NXT)/R32+ACC*Y(5,NXT)/VEL
        F(6,NXT)=-XMU*Y(3,NXT)/R32+ACC*Y(6,NXT)/VEL
        F(7,NXT)=0.0D+00
        F(8,NXT)=0.0D+00

C       END OF EQUATIONS OF MOTION
C       IS THIS ALL ?

        IF (MODE.EQ.0) RETURN

C       IT ISNT ALL ... CALCULATE THE EQUATIONS OF VARIATION
C       FIRST, CALCULATE A MATRIX .... ONLY LOWER 3X3 ISNT HARDWIRED

        R52=R32**(5.0D+00/3.0D+00)

C       DIAGONAL TERMS IN A MATRIX

        AM(4,1)=-XMU/R32+3.0D+00*XMU*Y(1,NXT)*Y(1,NXT)/R52
        AM(5,2)=-XMU/R32+3.0D+00*XMU*Y(2,NXT)*Y(2,NXT)/R52
```

```
      AM(6,3)=-XMU/R32+3.0D+00*XMU*Y(3,NXT)*Y(3,NXT)/R52

C     OFF DIAGONAL TERMS IN A MATRIX USE SYMMETRY TO AVOID
C     AS MUCH CALCULATION AS POSSIBLE...THIS POINT IS DEEP
C     WITHIN LOTS OF LOOPS!!!!

      AM(4,2)=3.0D+00*XMU*Y(1,NXT)*Y(2,NXT)/R52
      AM(5,1)=AM(4,2)
      AM(4,3)=3.0D+00*XMU*Y(1,NXT)*Y(3,NXT)/R52
      AM(6,1)=AM(4,3)
      AM(5,3)=3.0D+00*XMU*Y(2,NXT)*Y(3,NXT)/R52
      AM(6,2)=AM(5,3)

C     NOW SOME STUFF FOR THE OTHER TERMS

      AM(4,4)=-Y(4,NXT)*Y(4,NXT)*ACC/V32+ACC/VEL
      AM(5,5)=-Y(5,NXT)*Y(5,NXT)*ACC/V32+ACC/VEL
      AM(6,6)=-Y(6,NXT)*Y(6,NXT)*ACC/V32+ACC/VEL

      AM(4,5)=-Y(4,NXT)*Y(5,NXT)*ACC/V32
      AM(5,4)=AM(4,5)

      AM(4,6)=-Y(4,NXT)*Y(6,NXT)*ACC/V32
      AM(6,4)=AM(4,6)

      AM(5,6)=-Y(5,NXT)*Y(6,NXT)*ACC/V32
      AM(6,5)=AM(5,6)

      AM(4,7)=Y(4,NXT)*ACC/VVE
      AM(5,7)=Y(5,NXT)*ACC/VVE
      AM(6,7)=Y(6,NXT)*ACC/VVE

      VAT=ACC*ACC*T/VEM+ACC/MASS

      AM(4,8)=Y(4,NXT)*VAT/VEL
      AM(5,8)=Y(5,NXT)*VAT/VEL
      AM(6,8)=Y(6,NXT)*VAT/VEL

C     THE A MATRIX IS NOW CALCULATED

C     NOW, CALCULATE PHI DOT=A*PHI AND PUT INTO LAST
C     64 SLOTS OF THE F MATRIX

      DO 800 IRG=1,8
         DO 800 IRH=1,8
            IRI=8*IRH+IRG
            F(IRI,NXT)=0.0D+00
            DO 700 IRJ=1,8
               IRK=8*IRH+IRJ
               F(IRI,NXT)=F(IRI,NXT)+AM(IRG,IRJ)*Y(IRK,NXT)
700         CONTINUE
800   CONTINUE

C     PHI DOT=A*PHI IS NOW DONE
```

```
      RETURN
      END

      SUBROUTINE VEHD(VE,MDOT,MASSO,T,IRF,TEPOCH,TSTAGE)

      DOUBLE PRECISION VE,MDOT,MASSO,T,TEPOCH,TSTAGE

      INTEGER IRF

      DOUBLE PRECISION TIME,ISP,THRUST

      TIME=T*806.8118744D+00

      IF (IRF.EQ.1) THEN
         IF (TIME.LT.165.0D+00) THEN
            ISP=301.6D+00
            THRUST=531250.0D+00
            MASSO=410028.0D+00
            TSTAGE=TEPOCH
            VE=ISP/806.8118744D+00
            MDOT=THRUST/VE
         END IF
         IF ((TIME.LT.375.0D+00).AND.(TIME.GE.165.0D+00)) THEN
            ISP=318.0D+00
            THRUST=100700.0D+00
            MASSO=102028.0D+00
            TSTAGE=165.0D+00/806.8118744D+00
            VE=ISP/806.8118744D+00
            MDOT=THRUST/VE
         END IF
         IF ((TIME.LT.520.4D+00).AND.(TIME.GE.375.0D+00)) THEN
            ISP=295.0D+00
            THRUST=42200.0D+00
            MASSO=24028.0D+00
            TSTAGE=375.0D+00/806.8118744D+00
            VE=ISP/806.8118744D+00
            MDOT=THRUST/VE
         END IF
         IF (TIME .GT. 520.4) THEN
            MASSO=2628.0D+00
            TSTAGE=520.4D+00/806.8118744D+00
            VE=0.0D+00
            MDOT=0.0D+00
     END IF
      END IF

      RETURN
      END

      SUBROUTINE RAZEL(R,V,RHO,AZ,EL,TO,T,RS,TRM)

      DOUBLE PRECISION R(0:3),V(0:3),RHO,AZ,EL,TO,T,RS(0:3),TRM(3,3)

      DOUBLE PRECISION LAT,LON,LST,ZVEC(0:3),SVEC(0:3),EVEC(0:3),RAD,
     +   RHOVE(0:3),KVEC(0:3),RHOVEC(0:3),RE(0:3),RSE(0:3)
```

```
      INTEGER ING,INH,INI,INJ,INK,INL,INM,INN,INNL

      CHARACTER ANS

      IF (ING.EQ.0) THEN
         PRINT*,'ENTER SENSOR TYPE, LAND OR SPACE, IN QUOTES'
         READ*,ANS
         ING=10
         RAD=3.1415926535D+00/180.0D+00
         KVEC(1)=0.0D+00
         KVEC(2)=0.0D+00
         KVEC(3)=1.0D+00
      END IF

      IF ((ANS.EQ.'L').AND.(INH.EQ.0)) THEN
         PRINT*,'INPUT THE LAT AND LON OF SITE IN DEG, EAST+,WEST-'
         READ*,LAT,LON
         LAT=LAT*RAD
         LON=LON*RAD
         INH=10
      END IF

      CALL LSTIME(LST,T,TO,LON)

      CALL RADST(RS,LAT,LST,T,TO,ANS,INO)

      DO 100 INI=1,3
         RHOVE(INI)=R(INI)-RS(INI)
100   CONTINUE
      CALL MAG(RHOVE)
      RHO=RHOVE(0)

C     SET UP LOCAL COORDINATE SYSTEM

      DO 110 INJ=1,3
         ZVEC(INJ)=RS(INJ)/RS(0)
110   CONTINUE
      CALL CROSS(KVEC,ZVEC,EVEC)
      DO 112 INM=1,3
         EVEC(INM)=EVEC(INM)/EVEC(0)
112   CONTINUE
      CALL CROSS(EVEC,ZVEC,SVEC)
      DO 114 INN=1,3
         SVEC(INN)=SVEC(INN)/SVEC(0)
114   CONTINUE

C     SET UP THE TRANSFORMATION FOR IJK = TRM*SEZ

      DO 120 INL=1,3
         TRM(INL,1)=SVEC(INL)
         TRM(INL,2)=EVEC(INL)
         TRM(INL,3)=ZVEC(INL)
120   CONTINUE
      DO 121 INNL=1,3
```

```
            RE(INNL)=R(INNL)
            RSE(INNL)=RS(INNL)
121      CONTINUE

C        CONVERT TO SEZ FOR CALCULATIONS

         DO 130 INK=1,3
            RHOVEC(INK)=RHOVE(1)*TRM(1,INK)+RHOVE(2)*TRM(2,INK)
     +         +RHOVE(3)*TRM(3,INK)

C        NOTE!!! HERE WE DO NOT TRANSFORM R TO SEZ SINCE WE WILL NOT
C        BE CALCULATING H AS IN OBSER!!!!

            RS(INK)=RSE(1)*TRM(1,INK)+RSE(2)*TRM(2,INK)
     +         +RSE(3)*TRM(3,INK)
130      CONTINUE

         IF (RHOVEC(1).EQ.0.0D+00) THEN
            IF (RHOVEC(2).GT.0.0D+00) AZ=90.0D+00*RAD
            IF (RHOVEC(2).LT.0.0D+00) AZ=270.0D+00*RAD
            IF (RHOVEC(2).EQ.0.0D+00) THEN
               AZ=0.0D+00
               IF (RHOVEC(3).GT.0.0D+00) EL=90.0D+00*RAD
               IF (RHOVEC(3).LT.0.0D+00) EL=-90.0D+00*RAD
            END IF
         END IF

         IF ((RHOVEC(1).NE.0.0D+00).AND.(RHOVEC(2).NE.0.0D+00)) THEN
            AZ=DATAN(RHOVEC(2)/RHOVEC(1))
            EL=DATAN(RHOVEC(3)/DSQRT(RHOVEC(1)*RHOVEC(1)+RHOVEC(2)*
     +         RHOVEC(2)))
            IF (RHOVEC(1).LT.0.0D+00) AZ=AZ+180.0D+00*RAD
            IF ((RHOVEC(1).GT.0.0D+00).AND.(RHOVEC(2).LT.0.0D+00)) AZ=
     +         AZ+360.0D+00*RAD
         END IF

         RETURN
         END
$
```

APPENDIX B

A MATRIX

$$
\left[
\begin{array}{cccc}
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
-\dfrac{\mu}{r^3} + \dfrac{3\mu x^2}{r^5} & \dfrac{3\mu xy}{r^5} & \dfrac{3\mu xz}{r^5} & -\dfrac{v_x^2\, a}{v^3} + \dfrac{a}{v} \\
\dfrac{3\mu yx}{r^5} & -\dfrac{\mu}{r^3} + \dfrac{3\mu y^2}{r^5} & \dfrac{3\mu yz}{r^5} & -\dfrac{v_y v_x a}{v^3} \\
\dfrac{3\mu xz}{r^5} & \dfrac{3\mu yz}{r^5} & -\dfrac{\mu}{r^3} + \dfrac{3\mu z^2}{r^5} & -\dfrac{v_x v_z a}{v^3} \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{array}
\right.
$$

$$
\left.
\begin{array}{cccc}
0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
-\dfrac{v_x v_y a}{v^3} & -\dfrac{v_x v_z a}{v^3} & \dfrac{v_y a}{v V_e} & \dfrac{v_x}{v}\dfrac{a^2 t}{V_e M} + \dfrac{a}{M} \\
-\dfrac{v_y^2 a}{v^3} + \dfrac{a}{v} & -\dfrac{v_y v_z a}{v^3} & \dfrac{v_y a}{v V_e} & \dfrac{v_y}{v}\dfrac{a^2 t}{V_e M} + \dfrac{a}{M} \\
-\dfrac{v_y v_z a}{v^3} & -\dfrac{v_z^2 a}{v^3} + \dfrac{a}{v} & \dfrac{v_z a}{v V_e} & \dfrac{v_z}{v}\dfrac{a^2 t}{V_e M} + \dfrac{a}{M} \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{array}
\right]
$$

where

$A_{ij} = \partial \bar{F}_i / \partial \bar{x}_j$

x,y,z = positional components of $\bar{r}$ vector

v_x, v_y, v_z = velocity components of velocity vector

M = mass ratio ($\dot{m} / m_o$)

V_e = exhaust velocity

a = as defined in Equation (2-8)

APPENDIX C

NONLINEAR LEAST SQUARES ALGORITHM

1. Pick $\bar{x}_{ref}(t_o)$, initial guess for state vector

   - Set Q and read in data for all observations (G)

   - Initialize
     $\phi = I$
     $P^{-1} = 0$
     $T^T Q^{-1} \bar{r} = 0$

2. For each observation time

   - move $\bar{x}_{ref}(t_o)$ to $\bar{x}_{ref}(t_i)$

     - Haming and rhs do this for $\bar{x}_{ref}$, also ϕ

   - calculate predicted data using $\bar{x}_{ref}(t_i)$, $\bar{z}pred$

   - calculate residual $\bar{r}_i = \bar{z}_i - \bar{z}pred_i$

   - calculate H

   - calculate $T = H\phi$

   - sum $\sum T^T Q^{-1} T$

   - sum $\sum T^T Q^{-1} \bar{r}$

   loop back until all the data is processed

3. Calculations

   - $P = [\ T^T Q^{-1} T\ ]^{-1}$

   - $\delta \bar{x} = P\ T^T Q^{-1} \bar{r}$

   - update the $\bar{x}_{ref}(t_o) = \bar{x}_{ref}(t_o) + \delta \bar{x}(t_o)$

   - check convergence , $\delta \bar{x}(i) < [\ P_{ii}\ ]^{1/2}$

   - if good, end with $\bar{x}_{ref}(t_o)$

   - if not, begin at start with $\bar{x}_{ref}(t_o)$ and reset

     $\phi = I$; $T^T Q^{-1} \bar{r} = 0$; $P^{-1} = 0$; $t = t_o$
     $\bar{y}(*,1) = \bar{x}_{ref}(t_o)$

APPENDIX D

BAYES FILTER ALGORITHM

1. Pick $\bar{x}_{refu}(t_{epoch})$, initial guess for state vector

   set Q and $\bar{x}_{ref} = \bar{x}_{refu}$
   read all data and store in $\bar{z}$

   set $P^{-1}(-) = 0$ $\quad$ $t_{epoch} = 0$

2. For each Bayes iteration

3. For each least squares iteration

   Set $\phi = [I]$; $T^T Q^{-1} T = 0$; $T^T Q^{-1} \bar{r} = 0$

4. For each observation time

   move $\bar{x}_{ref}(t_{epoch})$ to $\bar{x}_{ref}(t_i)$
   Haming and rhs do this to $\bar{x}_{ref}$ and

   calculate predicted data using $\bar{x}_{ref}(t_i) = \bar{z}pred$

   calculate H

   T= H

   sum $\sum T^T Q^{-1} T$

   sum $\sum T^T Q^{-1} \bar{r}$

5. loop back until each data segment is processed

   $P^{-1}(+) = P^{-1}(-) + T^T Q^{-1} T$

   $\delta \bar{x} = P(+) [P^{-1}(-)(\bar{x}(-) - \bar{x}_{ref}) + T^T Q^{-1} \bar{r}]$

   $\bar{x}_{ref}(t_o) = \bar{x}_{ref}(t_o) + \delta \bar{x}$

   determine convergence as done for least squares

   no: look back for another L.S. iteration

   yes:

   $P^{-1}(-) = P^{-1}(+)$
   $\bar{x}_{ref}(t_i) = \bar{y}(*,nxt)$
   $\bar{x}_{refu}(t_i) = \bar{x}_{ref}(t_i)$
   $P^{-1}(-) = \phi^{-1T} P^{-1} \phi^{-1}$
   $t_{epoch} = t$

APPENDIX E

BAYES FILTER PROGRAMS

Description

The purpose of this program is to utilize the truth model data and estimate the launch vehicle parameters of the vehicle. It accomplishes this by using a Bayes Filter algorithm. The program is identical to the one Capt. Vallado developed in his thesis (reference 4) with the addition of a staging detection routine (subroutine Stage) and an additional estimation routine to aid in determining the staging time as well as the next stage exhaust velocity and mass ratio (V_e and M). A brief description of the subroutines peculiar to this program are as follows:

MMPY

This subroutine multiplies two matrices together and outputs the product.

MTRANS

This subroutine calculates the transpose of a matrix.

MATPRT

This subroutine prints a matrix.

OBSER

This subroutine calculates the observation relationships. Its main functions are to calculate the $\overline{G}$ matrix, the predicted value, the H matrix, and the in-track residual for the first pass of nonlinear least squares for each Bayes segment.

STAGE

This subroutine is responsible for utilizing the in-track residual computed by the Obser subroutine, and determining if a staging event has taken place. It then passes this information to the staging estimator in order to estimate the staging time and next stage vehicle parameters. Once the staging estimator has converged, the new values for the product of exhaust velocity and mass ratio and the time of the staging event are passed back to the main program in order to restart the Bayes Filter routine to process observation data for the next stage.

```
      PROGRAM BAYES

C     NONLINEAR LEASTSQUARES ALGORITHM

C     THIS PROGRAM ACCOMPLISHES A NONLINEAR LEAST SQUARES ALGORITHM
C     FOR THE PROBLEM OF ESTIMATION OF LAUNCH VEHICLE PERFORMANCE
C     PARAMETERS.  THE PROGRAM USES OBSER TO CALCULATE THE Q INVERSE,
C     THE APPROPRIATE H MATRIX, AND THE OBSERVATION MATRIX.  THE
C     PROGRAM ALSO USES DHAMING TO NUMERICALLY INTEGRATE THE STATE,
C     AND RHS TO CALCULATE THE EOM AND EOV.

C     THE COMMON TERMS

      COMMON /HAM/ T,Y(72,4),F(72,4),ERR(72),N,DT,MODE,TEPOCH,TSTAGE

      DOUBLE PRECISION T,Y,F,ERR,DT,TEPOCH,TSTAGE

      INTEGER N,MODE,NXT

C     THE OTHER TERMS

      DOUBLE PRECISION TIMEOB(500),RHO(500),AZ(500),EL(500),
     +        PHI(8,8),H(3,8),TMAT(3,8),Z(3),ZPRED(3),DX(8,1),
     +        Q1(3,3),RESID(3),TOB,WORK(8),HTQ1(8,3),PINVO(8,8),
     +        HTQ1R(8,1),XREF(8,1),P(8,8),TMATT(8,3),SIGTRK,
     +        PINVN(8,8),XREFU(8,1),XMXREF(8,1),PNDX(8,1),
     +        PNDXPH(8,1),HTQ1T(8,8),PINVNC(8,8),PC(8,8),PINVO1
     +        (8,8),PHIC(8,8),PHIIN(8,8),PHIT(8,8),BETA(8,8),
     +        BETAT(8,8),PINVO2(8,8),RINTRK(500),TIMTRK(500),
     +        XSTAGE(3),PSTG(2,2)

      INTEGER ILA,ILB,ILC,ILD,ILE,ILF,MAXIT,NOB,TROP,ILNOB,
     +        NPTS,ILCC,ISTAGE,ICOUNT,ILG,ILH,ILI,ILJ,ILK,ILL,
     +        ILM,ILN,ILO,ILP,NUMPTS

C     OPEN OUTPUT AND INPUT FILES FOR FILTER DATA, IN TRACK RESIDUALS
C     AND INPUT SENSOR DATA

      OPEN(UNIT=18,FILE='FILTDAT',ACCESS='SEQUENTIAL',STATUS='NEW')
      OPEN(UNIT=19,FILE='PLOTDAT',ACCESS='SEQUENTIAL',STATUS='NEW')
      OPEN(UNIT=14,FILE='TDATA',ACCESS='SEQUENTIAL',STATUS='OLD')
      REWIND(UNIT=14)

C     READ IN INITIAL DATA AND ALL CONTROL PARAMATERS

      PRINT*,'INPUT EPOCH TIME'
      READ*,TEPOCH

      PRINT*,'INPUT INITIAL STATE VECTOR GUESS, XREF'
      READ*,XREFU(1,1),XREFU(2,1),XREFU(3,1),XREFU(4,1),XREFU(5,1),
     +        XREFU(6,1),XREFU(7,1),XREFU(8,1)
```

```
      PRINT*,'INPUT THE MAX LS ITERATIONS'
      READ*,MAXIT

      PRINT*,'INPUT THE RANK OF P'
      READ*,TROP

      PRINT*,'INPUT THE NUMBER OF BAYES LOOP ITERATIONS'
      READ*,IBLOOP

      PRINT*,'INPUT # DATA POINTS READ EACH LEASTSQUARES RUN'
      READ*,ILNOB

      PRINT*,'INPUT BETA, FOR THE FADING MEMORY '
      READ*,BETA(1,1),BETA(2,2),BETA(3,3),BETA(4,4),BETA(5,5),
     +       BETA(6,6),BETA(7,7),BETA(8,8)

C     PRINT OUT THE INPUT

10    FORMAT(/,31X,'NONLINEAR BAYES FILTER',/,/,2X,
     +       'INITIAL STATE VECTOR : ',/,2X,4E18.11,/,2X,4E18.11,
     +       /,2X,'INITIAL TIME : ',F8.6,' # OF DATA POINTS : ',
     +       I4,/,2X,'MAX LS ITERATIONS :  ',I8,
     +       '  # OF BAYES CHUNKS :  ',I4,/,2X,
     +       'MAX BAYES ITERATIONS  :  ',I4,'  RANK OF P  : ',
     +       I11,/,2X,'BETA MATRIX = ',8F6.3)

      WRITE(18,10) XREFU,TEPOCH,NOB,MAXIT,ILNOB,IBLOOP,TROP,
     +          BETA(1,1),BETA(2,2),BETA(3,3),BETA(4,4),BETA(5,5),
     +          BETA(6,6),BETA(7,7),BETA(8,8)

C     SET LAST ITERATION, STAGING EVENT FLAG AND VARIOUS COUNTERS

      NDATA=3
      TSTAGE=TEPOCH
      ISTAGE=0
      IDONE=0
      CALL MEQL(XREFU,8,1,XREF)
      DO 40 IBJ=1,8
         DO 40 IBI=1,8
            PINVO(IBJ,IBI)=0.0D+00
40    CONTINUE

C     BEGIN BAYES FILTER LARGE LOOP

      DO 10000 IBG=1,IBLOOP

C     BEGIN ITERATION  LOOP - NONLINEAR LEAST SQUARES

C     READ IN OBSERVATIONS FOR EACH BAYES SEGMENT PROVIDING IT IS
C     NOT JUST AFTER A STAGING EVENT

         IF (ISTAGE.EQ.1) GOTO 35
         NUMPTS=ILNOB
```

```
      DO 30 ILB=1,ILNOB
     +   READ(14,*,END=7000) RHO(ILB),AZ(ILB),EL(ILB),TIMEOB(ILB)
30    CONTINUE

35    CONTINUE

      DT=TIMEOB(2)-TIMEOB(1)

      DO 9999 ILC=1,MAXIT

C     REINITIALIZE NUMERICAL INTEGRATION PARAMETERS

         T=TEPOCH
         MODE=1
         N=72

C     ICS ARE NEW REFERENCE TRAJ GUESS

         DO 50 ILD=1,8
            Y(ILD,1)=XREF(ILD,1)
50       CONTINUE

C     PHI INITIAL CONDITIONS

         DO 60 ILE=9,72
            Y(ILE,1)=0.0D+00
60       CONTINUE

         DO 70 ILF=9,72,9
            Y(ILF,1)=1.0D+00
70       CONTINUE

C     INITIALIZE HAMING AND RESET THE TIME

         NXT=0
         CALL HAMING(NXT)
         T=TEPOCH

C     INITIALIZE BUFFERS FOR MATRIX PRODUCT ACCUMULATION

         DO 80 ILG=1,8
            HTQ1R(ILG,1)=0.0D+00
            DO 80 ILH=1,8
               HTQ1T(ILG,ILH)=0.0D+00
80       CONTINUE

C     PRINT FIRST OR LAST PASS RESIDUAL HEADERS WHEN NECESSARY

90       FORMAT(/,2X,'FIRST PASS RESIDUALS: ',/)
95       FORMAT(/,2X,'LAST PASS RESIDUALS: ',/)
         IF(ILC.EQ.1) WRITE(18,90)
         IF(IDONE.EQ.1) WRITE(18,95)
```

```
C       OBSERVATION PROCESSING LOOP

              NPTS=0
              IF(ILC.EQ.1) SIGTRK=0.0D+00

              DO 1000 ILI=1,NUMPTS

C       EXTRACT EACH OBSERVATION

                 TOB=TIMEOB(ILI)
                 Z(1)=RHO(ILI)
                 Z(2)=AZ(ILI)
                 Z(3)=EL(ILI)

C       NUMERICALLY INTEGRATE STATE AND PHI TO OBS TIME
C       THE NUMBER OF STEPS HERE IS EQUAL TO 1 SINCE WE
C       HAVE DT SET EXACTELY THE SAME AS THE TRUTH DATA WE READ

                 NSTP=1
                 DO 100 ILK=1,NSTP
                    CALL HAMING(NXT)
100              CONTINUE

                 ILCC=ILC
                 NPTS=NPTS+1

C       OBTAIN MATRICES FOR THIS OBSERVATION

                  CALL OBSER(TOB,Q1,ZPRED,H,NXT,Z,RINTRK,NPTS,ILCC,
     +                  TIMTRK,SIGTRK)

C       MATRIX STUFF - THIS OBSERVATION

                 DO 120 ILL=1,NDATA
                    RESID(ILL)=Z(ILL)-ZPRED(ILL)
120              CONTINUE

                 IF(ILI.LT.5) GOTO 200
                 IF((IDONE.EQ.1).AND.(ILI.LT.5)) GOTO 200
                 IF((IDONE.EQ.1).AND.(ILI.GE.5)) GOTO 240
                 GOTO 250
200              CONTINUE
                 WRITE(18,*)'TIME, RES =',TOB,(RESID(ILM),ILM=1,NDATA)

C       IF THIS IS LAST PASS, WE'VE ALREADY CONVERGED,
C       SO SKIP MATRIX CALCULATIONS

240              IF(IDONE.EQ.1) GOTO 9000
250              CONTINUE

C       EXTRACT PHI MATRIX IN NORMAL FORM

                 DO 260 ILN=1,8
```

```
                  DO 270 ILO=1,8
                     PHI(ILN,ILO)=Y(8*ILO+ILN,NXT)
270               CONTINUE
260             CONTINUE

C       FORM MATRIX ****** TMAT=H*PHI

                CALL MMPY(H,3,8,PHI,8,TMAT)

C       FORM MATRIX ******  HTQ1=T TRANSPOSE * Q INVERSE

                CALL MTRANS(TMAT,3,8,TMATT)
                CALL MMPY(TMATT,8,3,Q1,3,HTQ1)

C       FORM MATRIX ****** HTQ1T=T TRANSPOSE Q INVERSE T
C       SUM THROUGH THE OBSERVATIONS

                DO 290 ILP=1,8
                  DO 290 ILQ=1,8
                    DO 280 ILR=1,NDATA
                      HTQ1T(ILP,ILQ)=HTQ1T(ILP,ILQ)+HTQ1(ILP,ILR)
     +                                *TMAT(ILR,ILQ)
280                 CONTINUE
290             CONTINUE

C       FORM MATRIX ****** HTQ1R=T TRANSPOSE Q INVERSE R
C       SUM THROUGH THE OBSERVATIONS

                DO 150 ILS=1,8
                  DO 150 ILT=1,NDATA
                    HTQ1R(ILS,1)=HTQ1R(ILS,1) +HTQ1(ILS,ILT)*
     +                              RESID(ILT)
150             CONTINUE
9000            CONTINUE

1000          CONTINUE

C       LOOK BACK FOR OBSERVATION LOOP OF LEAST SQUARES

              DO 420 IBE=1,8
                DO 420 IBF=1,8
                  PINVN(IBE,IBF)=PINVO(IBE,IBF)+HTQ1T(IBE,IBF)
420           CONTINUE

              CALL MEQL(PINVN,8,8,PINVNC)

C       HAVE WE JUST FINISHED PRINTING LAST PASS RESIDUALS ?

              IF(IDONE.EQ.1) GOTO 5000

C       NOW WE CHECK FOR IN OR OUT OF TRACK CONDITION
C       IF WE ARE NOT IN THE FIRST BAYES FILTER SEGMENT
C       OR HAVE JUST PAST A STAGING EVENT
```

```
                 IF((IBG.EQ.1).OR.(ISTAGE.EQ.1)) THEN
                    ISTAGE=0
                    GOTO 6000
                 END IF

C       CHECK IF THIS IS THE FIRST ITERATION OF THE BAYES SEGMENT
C       BEFORE STARTING THE STAGING ROUTINE

                 IF(ILC.EQ.1) CALL STAGE(RINTRK,ILNOB,TIMTRK,SIGTRK,
     +                             ISTAGE,XSTAGE,XREFU,TSTAGE,
     +                             PSTG)

C       IF THE VEHICLE HAS STAGED THE STAGING ROUTINE IS PERFORMED

                 IF (ISTAGE.EQ.1) THEN

C       STAGING ROUTINE

C       FIRST IDENTIFY THE POINT IN THE DATA WHERE STAGING TOOK PLACE

                    ICOUNT=0
                    DO 500 ILG=1,ILNOB
                       IF (XSTAGE(2).LT.TIMEOB(ILG)) GOTO 505
                       ICOUNT=ICOUNT+1
500                 CONTINUE

505                 CONTINUE

C       NOW THAT THE POINT IN THE SENSOR DATA DIRECTLY BEFORE STAGING
C       HAS BEEN IDENTIFIED, THE STATE AND COVARIANCE MUST BE MOVED
C       TO THE STAGING TIME

C       REINITIALIZE INTEGRATION PARAMETERS

                    T=TEPOCH
                    MODE=1
                    N=72

C       SET THE STATE INITIAL CONDITIONS TO THE LAST GOOD VALUES
C       OF THE STATE AVAILABLE

                    DO 510 ILH=1,8
                       Y(ILH,1)=XREFU(ILH,1)
510                 CONTINUE

C       SET INITIAL CONDITIONS FOR THE PHI MATRIX

                    DO 515 ILI=9,72
                       Y(ILI,1)=0.0D+00
515                 CONTINUE

                    DO 520 ILJ=9,72,9
```

```
                    Y(ILJ,1)=1.0D+00
520              CONTINUE

C       INITIALIZE HAMING AND RESET THE TIME

                 NXT=0
                 CALL HAMING (NXT)
                 T=TEPOCH

C       PROPAGATE STATE AND PHI TO THE STAGE TIME

                 DO 525 ILK=1,ICOUNT
                    CALL HAMING (NXT)
525              CONTINUE

                 TEPOCH=T

C       EXTRACT THE PHI MATRIX IN NORMAL FORM

                 DO 535 ILM=1,8
                    DO 534 ILN=1,8
                       PHI(ILM,ILN)=Y(8*ILN+ILM,NXT)
534                 CONTINUE
535              CONTINUE

C       CALCULATE UPDATED P MATRIX AT NEW START TIME

                 CALL MTRANS (PHI,8,8,PHIT)
                 CALL MMPY (PHI,8,8,P,8,PINV01)
                 CALL MMPY (PINV01,8,8,PHIT,8,P)

C       NEW GUESS FOR NEXT STAGE WILL BE COMPRISED OF POSITIONAL AND
C       VELOCITY DATA AT TIME INCREMENT JUST BEFORE STAGING EVENT,
C       VE WILL BE GIVEN VALUE OF LAST STAGE VE, AND M WILL BE GIVEN
C       VALUE OF ESTIMATED VE*M DIVIDED BY VE GUESS. WILL ALSO UPDATE
C       THE TSTAGE TIME.

                 DO 530 ILL=1,8
                    Y(ILL,1)=Y(ILL,NXT)
                    XREF(ILL,1)=Y(ILL,NXT)
530              CONTINUE

                 XREF(8,1)=XSTAGE(1)/XREF(7,1)
                 CALL MEQL (XREF,8,1,XREFU)
                 TSTAGE=XSTAGE(2)

C       MUST NOW ALTER THE COVARIANCE MATRIX TO TELL THE FILTER THAT
C       IT NO LONGER KNOWS THE VALUES OF VE AND M AS WELL AS BEFORE
C       STAGING EVENT

                 P(8,8)=PSTG(1,1)/(XREFU(7,1)*XREFU(7,1))+
     +                     (XSTAGE(1)*XSTAGE(1)/(XREFU(7,1)*
     +                     XREFU(7,1)*XREFU(7,1)*XREFU(7,1))*
```

```
     +                     P(7,7))

                 P(7,7)=2.0D+00*P(7,7)

C        PRINT HEADER FOR THE NEXT STAGE ESTIMATION

550              FORMAT (/,/,26X,'BEGIN ESTIMATION OF THE NEXT STAGE',
     +                  /,/,26X,'STAGING OCCURED AT : ',F8.4,2X,
     +                  'SECONDS',/,/)

                 WRITE (18,550) TSTAGE*806.8136D+00
                 WRITE (18,*) '  AT THE START OF THE NEXT STAGE THE'
                 WRITE (18,950) P

555              FORMAT (2X,'INITIAL STATE VECTOR IS:',/,/,/,2X,
     +                  4E18.11,/,2X,4E18.11)

                 WRITE (18,555) XREFU

C        FIRST MOVE THE REMAINING DATA POINTS TO THE FRONT OF THE
C        BAYES SEGMENT ARRAY

                 DO 560 ILO=1,ILNOB-ICOUNT
                    RHO(ILO)=RHO(ICOUNT+ILO)
                    AZ(ILO)=AZ(ICOUNT+ILO)
                    EL(ILO)=EL(ICOUNT+ILO)
                    TIMEOB(ILO)=TIMEOB(ICOUNT+ILO)
560              CONTINUE

C        CHANGE THE NUMBER OF POINTS TO BE PROCESSED AFTER A STAGING
C        EVENT BY 4 TIMES

                 NUMPTS=4*ILNOB

C        NOW READ IN ENOUGH OBSERVATION DATA POINTS TO FILL THE
C        RADAR OBSERVATION ARRAY FOR THE NEXT BAYES RUN

                 DO 570 ILP=ILNOB-ICOUNT+1,NUMPTS
                    READ (14,*,END=7000) RHO(ILP),AZ(ILP),EL(ILP),
     +                    TIMEOB(ILP)
570              CONTINUE

C        UPDATE THE COVARIANCE MATRIX TO PASS BACK TO BAYES FILTER

                 CALL MEQL(P,8,8,PC)
                 CALL LINV1F (PC,8,8,PINVO,0,WORK,IER)


                 GOTO 10000

              END IF

6000          CONTINUE
```

```
C       DATA IS PROCESSED....IMPROVE ESTIMATE
C       INVERT MATRIX H TRANSPOSE Q INVERSE H TO FIND
C       COVARIANCE P

              CALL LINV1F(PINVNC,TROP,8,P,0,WORK,IER)
              CALL MEQL(P,8,8,PC)

C       FORM MATRIX ****** DX=P*T TRANSPOSE Q INVERSE R

              DO 600 IBC=1,8
                 XMXREF(IBC,1)=XREFU(IBC,1)-XREF(IBC,1)
600           CONTINUE
              CALL MMPY(PINVO,8,8,XMXREF,1,PNDX)
              DO 640 IBD=1,8
                 PNDXPH(IBD,1)=PNDX(IBD,1)+HTQ1R(IBD,1)
640           CONTINUE
              CALL MMPY(P,8,8,PNDXPH,1,DX)

C       ADD IN STATE CORRECTIONS

              DO 700 ILV=1,8
                 XREF(ILV,1)=XREF(ILV,1)+DX(ILV,1)
700           CONTINUE

C       PRINT ITERATION, AND CURRENT GUESS

720           FORMAT(/,2X,'ITERATION ',I3,/,/,2X,'STATE CORRECTIONS'
     +               ,/,2X,4E18.11,/,2X,4E18.11)
              WRITE(18,720) ILC,DX
740           FORMAT(/,2X,'CURRENT REFERENCE TRAJECTORY STATE VECTOR ',
     +               'AT EPOCH: ',/,2X,4E18.11,/,2X,4E18.11,/)
              WRITE(18,740) XREF

C       SUCCESS ?????????
C       CHECK CONVERGENCE

              IFAIL=0
              DO 800 ILU=1,8
                 IF(DABS(DX(ILU,1)).GT.0.1*DSQRT(DABS(P(ILU,ILU))))
     +              IFAIL=1
800           CONTINUE

              IF (IFAIL .EQ. 0) IDONE=1

9999        CONTINUE

C       LOOP BACK FOR NEXT ITERATION OF LEAST SQUARES

C       FAILURE FOR THE LEAST SQUARES !!!!!!!!!!!!!!

900         FORMAT(2X,'MAXIMUM ITERATION LIMIT EXCEEDED WITHOUT
     +            CONVERGENCE.')
```

```
         PRINT 900
         WRITE(18,900)
         STOP

C       SUCCESS FOR THE LEAST SQUARES !!!!!!!!!!!!!!

5000     CONTINUE

940      FORMAT(/,2X,'CONVERGENCE ACHIEVED.',/,2X,
     +         'IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA',/,
     +         2X,'DECLARIUM EST ESTIMATIA',/)
         PRINT 940
         WRITE(18,940)

C       PRINT COVARIANCE MATRIX

950      FORMAT(/,2X,'COVARIANCE MATRIX AT EPOCH IS: ',/,
     +              8(1X,5E14.7,/,1X,3E14.7,/,/) )
         WRITE(18,950) P

C       LOAD NEW STATE VECTOR, AND RESET PHI

         DO 960 IBR=1,8
           Y(IBR,1)=Y(IBR,NXT)
           XREF(IBR,1)=Y(IBR,NXT)
960      CONTINUE
         CALL MEQL(XREF,8,1,XREFU)

C       EXTRACT PHI MATRIX IN NORMAL FORM

         DO 985 IBV=1,8
           DO 985 IBW=1,8
             PHI(IBV,IBW)=Y(8*IBW+IBV,NXT)
985      CONTINUE

C       CALCULATE UPDATED P MATRIX AT NEW START TIME

         CALL MEQL(PHI,8,8,PHIC)
         CALL LINV1F(PHIC,8,8,PHIIN,0,WORK,IER)
         CALL MTRANS(PHIIN,8,8,PHIT)
         CALL MMPY(PHIT,8,8,PINVN,8,PINVO1)
         CALL MMPY(PINVO1,8,8,PHIIN,8,PINVN)
         CALL MMPY(BETA,8,8,PINVN,8,PINVO)
         CALL MTRANS(BETA,8,8,BETAT)
         CALL MEQL(PINVO,8,8,PINVO2)
         CALL MMPY(PINVO2,8,8,BETAT,8,PINVO)

         TEPOCH =T
         IDONE=0
         WRITE(18,*) 'BEGIN NEXT BAYES LOOP'

10000   CONTINUE
```

```
C       LOOP BACK FOR BAYES FILTER LOOP

        WRITE(18,*) 'WE DID IT, SUCCESS WITH BAYES'

        GOTO 8000

7000    CONTINUE
        PRINT*,'RAN OUT OF RADAR OBSERVATIONS'
        WRITE(18,*)'  RAN OUT OF RADAR OBSERVATIONS'

8000    CONTINUE

        ENDFILE(UNIT=14)
        ENDFILE(UNIT=18)
        ENDFILE(UNIT=19)

        END

        SUBROUTINE MEQL(MAT7,MAT7R,MAT7C,MAT8)

        DOUBLE PRECISION MAT7(MAT7R,MAT7C)

        INTEGER MAT7R,MAT7C

        DOUBLE PRECISION MAT8(MAT7R,MAT7C)

        INTEGER IMF,IMG

        DO 3000 IMF=1,MAT7R
           DO 3000 IMG=1,MAT7C
              MAT8(IMF,IMG)=MAT7(IMF,IMG)
3000    CONTINUE

        RETURN
        END

        SUBROUTINE LSTIME(LST,T,TO,LON)

        DOUBLE PRECISION LST,T,TO,LON

        DOUBLE PRECISION THTGO,TWOPI,GST

        TO=0.0D+00
        TWOPI=6.28318530718D+00
        THTGO=98.85481D+00*(3.14159265359D+00/180.00D+00)
        GST=THTGO+1.0027379093D+00*((T*13.44686457D+00/
     +     1440.0D+00)-TO)
        GST=DMOD(GST,TWOPI)
        LST=GST+LON
        LST=DMOD(LST,TWOPI)

        RETURN
        END
```

```
      SUBROUTINE RADST(RS,LAT,LST,T,TO,ANS,INO)

      DOUBLE PRECISION RS(0:3),LAT,LST,T,TO

      CHARACTER ANS

      DOUBLE PRECISION STA,STE,STI,STOMGA,STARGP,STV(0:3),STM,STN,
     +   STNUO

      INTEGER INO

C     LAND BASED SENSOR

      IF (ANS.EQ.'L') THEN
         IF (INO.EQ.0) THEN
            PRINT*,'INPUT THE ELEVATION OF THE SITE'
            READ*,RS(0)
            INO=10
         END IF
         RS(1)=RS(0)*DCOS(LAT)*DCOS(LST)
         RS(2)=RS(0)*DCOS(LAT)*DSIN(LST)
         RS(3)=RS(0)*DSIN(LAT)

         RETURN

      END IF

C     SPACE BASED SENSOR

      IF (ANS.EQ.'S') THEN
         IF (INO.EQ.0) THEN
            PRINT*,'INPUT THE TRACKING SAT ORBIT DATA, A, E, I, OMEGA, ARGP'
            READ*,STA,STE,STI,STOMGA,STARGP
         END IF
         STN=DSQRT(1/(STA*STA*STA))
         STM=STN*(T-TO)
         CALL RANDV(STA,STE,STI,STOMGA,STARGP,STNUO,STM,RS,STV)
         CALL MAG(RS)
         IF (INO.EQ.0) THEN
64          FORMAT(3X,'A',6X,'E',6X,'I',5X,'OMEGA',3X,'ARGP',4X,'M')
66          FORMAT(6(1X,F6.3))
            WRITE(17,*)
            WRITE(17,*) 'THE TRACKING SATELLITE DATA IS'
            WRITE(17,64)
            WRITE(17,66) STA,STE,STI,STOMGA,STARGP,STM
            INO=10
         END IF

      END IF

      RETURN
      END
```

```
      SUBROUTINE RANDV(A,E,INC,OMEGA,ARGP,NUO,M,R,V)

      DOUBLE PRECISION A,E,INC,OMEGA,ARGP,NUO,M,R(0:3),V(0:3)

      DOUBLE PRECISION RAD,P,EL,EO,MO

      RAD=3.14159265359D+00/180.00D+00
      MO=M*RAD
      INC=INC*RAD
      ARGP=ARGP*RAD
      OMEGA=OMEGA*RAD
      P=A*(1-E*E)

C     NEWTON RHAPSON ITERATION

      EL=MO
8     EO=EL
      EL=EO-(EO-E*DSIN(EO)-MO)/(1.0D+00-E*DCOS(EO))
      IF (DABS(EL-EO).GT.1.0D-12) THEN
         EL=EO-(EO-E*DSIN(EO)-MO)/(1.0D+00-E*DCOS(EO))
         GOTO 8
      END IF

C     FIND THE VALUE OF THE TRUE ANOMALY

      NUO=DATAN2((DSQRT(1.0D+00-E*E))*DSIN(EL)/(1.0D+00-E*
     +   DCOS(EL)),(E-DCOS(EL))/(E*DCOS(EL)-1.0D+00))

C     POSITION AND VELOCITY VECTORS

      R(1)=P*DCOS(NUO)/(1.0D+00+E*DCOS(NUO))
      R(2)=R(1)*DTAN(NUO)
      R(3)=0.0D+00
      V(1)=-DSIN(NUO)/DSQRT(P)
      V(2)=(E+DCOS(NUO))/DSQRT(P)
      V(3)=0.0D+00

      RETURN
      END

      SUBROUTINE MAG(RX)

      DOUBLE PRECISION RX(0:3)

      RX(0)=DSQRT(RX(1)*RX(1)+RX(2)*RX(2)+RX(3)*RX(3))

      RETURN
      END

      SUBROUTINE CROSS(RIN,VIN,VX)

      DOUBLE PRECISION RIN(0:3),VIN(0:3),VX(0:3)
```

```
      VX(1)=RIN(2)*VIN(3)-VIN(2)*RIN(3)
      VX(2)=-RIN(1)*VIN(3)+VIN(1)*RIN(3)
      VX(3)=RIN(1)*VIN(2)-VIN(1)*RIN(2)
      CALL MAG(VX)

      RETURN
      END

      SUBROUTINE HAMING(NXT)

      COMMON /HAM/T,Y(72,4),F(72,4),ERREST(72),N,DT,MODE,TEPOCH,
     +          TSTAGE

      DOUBLE PRECISION T,Y,F,ERREST,DT,TEPOCH,TSTAGE

      INTEGER N,MODE,NXT

      INTEGER IDA,IDB,IDC,IDD,IDE,IDF,IDG,IDH,IDI,IDJ,IDK,IDL,IDM,IDN

      DOUBLE PRECISION TOL,HH,XO

C     THE VARIABLES ARE USED AS FOLLOWS
C     T                  INDEPENDENT VARIABLE (TIME)
C     Y(72,4)            STATE VECTOR IN 4 COPIES
C     F(72,4)            EQUATIONS OF MOTION, 4 COPIES
C                           CALL RHS(NXT) UPDATES ENTRY IN F
C     ERREST             ESTIMATE OF TRUNCATION ERROR
C     N                  NUMBER OF EQUATIONS BEING INTEGRATED
C     DT                 TIME STEP
C     MODE               0 FOR EOM.  1 FOR EOM AND EOV

      TOL=1.0D-12
      IF (NXT) 190,10,200

C     SWITCH ON STARTING ALGORITHM OR NORMAL PROPOGATION
C     THIS IS HAMINGS STARTING ALGORITHM...A PREDICTOR-CORRECTOR
C     NEEDS 4 VALUES OF THE STATE VECTOR , AND YOU ONLY HAVE 1, THE
C     I.C.  HAMING USES PRICARD ITERATION (SLOW AND PAINFULL) TO GET
C     THE OTHER THREE.
C     IF IT FAILS, NXT= 0 ON EXIT, OTHERWISE, NXT=1, AND IT'S OK.

10    XO=T
      HH=DT/2.0D+00
      CALL RHS(1)
      DO 40 IDA=2,4
         T=T+HH
         DO 20 IDB=1,N
            Y(IDB,IDA)=Y(IDB,IDA-1)+HH*F(IDB,IDA-1)
20       CONTINUE
         CALL RHS(IDA)
         T=T+HH
         DO 30 IDC=1,N
```

```
             Y(IDC,IDA)=Y(IDC,IDA-1)+DT*F(IDC,IDA)
 30        CONTINUE
           CALL RHS(IDA)
 40     CONTINUE
        IDD=-10
 50     IDE=1
        DO 120 IDF=1,N
           HH=Y(IDF,1)+DT*(9.0D+00*F(IDF,1)+19.0D+00*F(IDF,2)-
     +       5.0D+00*F(IDF,3)+F(IDF,4))/24.0D+00
           IF (DABS(HH-Y(IDF,2)).LT.TOL) GOTO 70
           IDE=0
 70        Y(IDF,2)=HH
           HH=Y(IDF,1)+DT*(F(IDF,1)+4.0D+00*F(IDF,2)+F(IDF,3))/3.0D+00
           IF (DABS(HH-Y(IDF,3)).LT.TOL) GOTO 90
           IDE=0
 90        Y(IDF,3)=HH
           HH=Y(IDF,1)+DT*(3.0D+00*F(IDF,1)+9.0D+00*F(IDF,2)+
     +        9.0D+00*F(IDF,3)+3.0D+00*F(IDF,4))/8.0D+00
           IF (DABS(HH-Y(IDF,4)).LT.TOL) GOTO 110
           IDE=0
 110       Y(IDF,4)=HH
 120    CONTINUE
        T=X0
        DO 130 IDG=2,4
           T=T+DT
           CALL RHS(IDG)
 130    CONTINUE
        IF (IDE) 140,140,150
 140    IDD=IDD+1
        IF (IDD) 50,280,280
 150    T=X0
        IDE=1
        IDD=1
        DO 160 IDH=1,N
           ERREST(IDH)=0.0
 160    CONTINUE
        NXT=1
        GOTO 280
 190    IDD=2
        NXT=IABS(NXT)

 C      THIS IS HAMINGS NORMAL PROPAGATION LOOP

 200    T=T+DT
        IDL=MOD(NXT,4)+1
        GOTO (210,230),IDE

 C      PERMUTE THE INDEX NXT MODULO 4

 210    GOTO (270,270,270,220),NXT
 220    IDE=2
 230    IDI=MOD(IDL,4)+1
        IDJ=MOD(IDI,4)+1
```

```
      IDK=MOD(IDJ,4)+1

C     THIS IS THE PREDICTOR PART

      DO 240 IDM=1,N
         F(IDM,IDI)=Y(IDM,IDL)+4.0D+00*DT*(2.0D+00*F(IDM,IDK)-
     +      F(IDM,IDJ)+2.0D+00*F(IDM,IDI))/3.0D+00
         Y(IDM,IDL)=F(IDM,IDI)-0.925619835D+00*ERREST(IDM)
240   CONTINUE

C     NOW THE CORRECTOR - FIX UP THE EXTRAPOLATED STATE
C     BASED ON THE BETTER VALUE OF THE EQUATIONS OF MOTION

      CALL RHS(IDL)
      DO 250 IDN=1,N
         Y(IDN,IDL)=(9.0D+00*Y(IDN,IDK)-Y(IDN,IDI)+3.0D+00*DT*
     +      (F(IDN,IDL)+2.0D+00*F(IDN,IDK)-F(IDN,IDJ)))/8.0D+00
         ERREST(IDN)=F(IDN,IDI)-Y(IDN,IDL)
         Y(IDN,IDL)=Y(IDN,IDL)+0.0743801653D+00*ERREST(IDN)
250   CONTINUE
      GOTO (260,270),IDD
260   CALL RHS(IDL)
270   NXT=IDL

280   RETURN
      END

      SUBROUTINE RHS(NXT)

      COMMON /HAM/T,Y(72,4),F(72,4),ERR(72),N,DT,MODE,TEPOCH,TSTAGE

      DOUBLE PRECISION T,Y,F,ERR,DT,TEPOCH,TSTAGE

      INTEGER N,MODE,NXT

      INTEGER IRA,IRB,IRC,IRD,IRG,IRH,IRI,IRJ,IRK

      DOUBLE PRECISION R32,V32,VEL,VAT,VVE,VEM,MASS,ACC,R52,AM(8,8),
     +   MASS0,MDOT,VE

C     THIS DATA STATEMENT HARDWIRES THE PARTS OF THE
C     A MATRIX WHICH ARE NEVER CHANGED ...ONLY THE MIDDLE
C     3 ROWS CHANGE EACH TIME

      DO 10 IRA=1,8
         DO 10 IRB=1,3
            AM(IRB,IRA)=0.0D+00
10    CONTINUE
      DO 20 IRC=1,8
         DO 20 IRD=7,8
            AM(IRD,IRC)=0.0D+00
20    CONTINUE
      AM(1,4)=1.0D+00
```

```
      AM(2,5)=1.0D+00
      AM(3,6)=1.0D+00

C     THE BASIC FUNCTION OF RHS IS TO CALCULATE THE EQUATIONS
C     OF MOTION (THE F ENTRIES) FROM THE GIVEN CURRENT STATE
C     (STORED IN Y) AND THE TIME T

C     EVALUATION OF THE EQUATIONS OF MOTION

C     REFERENCE BATES MEULLER & WHITE, PG 10, N BODY PROBLEM
C     WITH ORIGIN IN SUN.

C     POSITION DOT = VELOCITY VECTOR

      F(1,NXT)=Y(4,NXT)
      F(2,NXT)=Y(5,NXT)
      F(3,NXT)=Y(6,NXT)

C     VELOCITY DOT = GRAVITY ACCELERATION

      R32=(Y(1,NXT)*Y(1,NXT)+Y(2,NXT)*Y(2,NXT)+
     +   Y(3,NXT)*Y(3,NXT))**1.5D+00
      VEL=(Y(4,NXT)*Y(4,NXT)+Y(5,NXT)*Y(5,NXT)+
     +   Y(6,NXT)*Y(6,NXT))**.5D+00
      V32=VEL**3.0D+00

C     SET THE CONSTANTS WHICH WILL BE USED IN THE A MATRIX

      XMU=1.0D+00
      VE=Y(7,NXT)
      MASS=Y(8,NXT)
      VVE=VEL*VE
      VEM=VE*MASS
      ACC=VE*MASS/(1.0D+00-MASS*(T-TSTAGE))

6     F(4,NXT)=-XMU*Y(1,NXT)/R32+ACC*Y(4,NXT)/VEL
      F(5,NXT)=-XMU*Y(2,NXT)/R32+ACC*Y(5,NXT)/VEL
      F(6,NXT)=-XMU*Y(3,NXT)/R32+ACC*Y(6,NXT)/VEL
      F(7,NXT)=0.0D+00
      F(8,NXT)=0.0D+00

C     END OF EQUATIONS OF MOTION
C     IS THIS ALL ?

      IF (MODE.EQ.0) RETURN

C     IT ISNT ALL ... CALCULATE THE EQUATIONS OF VARIATION
C     FIRST, CALCULATE A MATRIX .... ONLY LOWER 3X3 ISNT HARDWIRED

      R52=R32**(5.0D+00/3.0D+00)

C     DIAGONAL TERMS IN A MATRIX
```

```
      AM(4,1)=-XMU/R32+3.0D+00*XMU*Y(1,NXT)*Y(1,NXT)/R52
      AM(5,2)=-XMU/R32+3.0D+00*XMU*Y(2,NXT)*Y(2,NXT)/R52
      AM(6,3)=-XMU/R32+3.0D+00*XMU*Y(3,NXT)*Y(3,NXT)/R52

C     OFF DIAGONAL TERMS IN A MATRIX USE SYMMETRY TO AVOID
C     AS MUCH CALCULATION AS POSSIBLE...THIS POINT IS DEEP
C     WITHIN LOTS OF LOOPS!!!!

      AM(4,2)=3.0D+00*XMU*Y(1,NXT)*Y(2,NXT)/R52
      AM(5,1)=AM(4,2)
      AM(4,3)=3.0D+00*XMU*Y(1,NXT)*Y(3,NXT)/R52
      AM(6,1)=AM(4,3)
      AM(5,3)=3.0D+00*XMU*Y(2,NXT)*Y(3,NXT)/R52
      AM(6,2)=AM(5,3)

C     NOW SOME STUFF FOR THE OTHER TERMS

      AM(4,4)=-Y(4,NXT)*Y(4,NXT)*ACC/V32+ACC/VEL
      AM(5,5)=-Y(5,NXT)*Y(5,NXT)*ACC/V32+ACC/VEL
      AM(6,6)=-Y(6,NXT)*Y(6,NXT)*ACC/V32+ACC/VEL

      AM(4,5)=-Y(4,NXT)*Y(5,NXT)*ACC/V32
      AM(5,4)=AM(4,5)

      AM(4,6)=-Y(4,NXT)*Y(6,NXT)*ACC/V32
      AM(6,4)=AM(4,6)

      AM(5,6)=-Y(5,NXT)*Y(6,NXT)*ACC/V32
      AM(6,5)=AM(5,6)

      AM(4,7)=Y(4,NXT)*ACC/VVE
      AM(5,7)=Y(5,NXT)*ACC/VVE
      AM(6,7)=Y(6,NXT)*ACC/VVE

      VAT=ACC*ACC*T/VEM+ACC/MASS

      AM(4,8)=Y(4,NXT)*VAT/VEL
      AM(5,8)=Y(5,NXT)*VAT/VEL
      AM(6,8)=Y(6,NXT)*VAT/VEL

C     THE A MATRIX IS NOW CALCULATED

C     NOW, CALCULATE PHI DOT=A*PHI AND PUT INTO LAST
C     64 SLOTS OF THE F MATRIX

      DO 800 IRG=1,8
         DO 800 IRH=1,8
            IRI=8*IRH+IRG
            F(IRI,NXT)=0.0D+00
            DO 700 IRJ=1,8
               IRK=8*IRH+IRJ
               F(IRI,NXT)=F(IRI,NXT)+AM(IRG,IRJ)*Y(IRK,NXT)
700         CONTINUE
```

```
800     CONTINUE

C       PHI DOT=A*PHI IS NOW DONE

        RETURN
        END

        SUBROUTINE RAZEL(R,V,RHO,AZ,EL,TO,T,RS,TRM)

        DOUBLE PRECISION R(0:3),V(0:3),RHO,AZ,EL,TO,T,RS(0:3),TRM(3,3)

        DOUBLE PRECISION LAT,LON,LST,ZVEC(0:3),SVEC(0:3),EVEC(0:3),RAD,
     +    RHOVE(0:3),KVEC(0:3),RHOVEC(0:3),RE(0:3),RSE(0:3)

        INTEGER ING,INH,INI,INJ,INK,INL,INM,INN,INNL

        CHARACTER ANS

        IF (ING.EQ.0) THEN
           PRINT*,'ENTER SENSOR TYPE, LAND OR SPACE, IN QUOTES'
           READ*,ANS
           ING=10
           RAD=3.14159265359D+00/180.0D+00
           KVEC(1)=0.0D+00
           KVEC(2)=0.0D+00
           KVEC(3)=1.0D+00
        END IF

        IF ((ANS.EQ.'L').AND.(INH.EQ.0)) THEN
           PRINT*,'INPUT THE LAT AND LON OF SITE IN DEG, EAST+,WEST-'
           READ*,LAT,LON
           LAT=LAT*RAD
           LON=LON*RAD
           INH=10
        END IF

        CALL LSTIME(LST,T,TO,LON)

        CALL RADST(RS,LAT,LST,T,TO,ANS,INO)

        DO 100 INI=1,3
           RHOVE(INI)=R(INI)-RS(INI)
100     CONTINUE
        CALL MAG(RHOVE)
        RHO=RHOVE(0)

C       SET UP LOCAL COORDINATE SYSTEM

        DO 110 INJ=1,3
           ZVEC(INJ)=RS(INJ)/RS(0)
110     CONTINUE
        CALL CROSS(KVEC,ZVEC,EVEC)
        DO 112 INM=1,3
```

```
         EVEC(INM)=EVEC(INM)/EVEC(0)
 112  CONTINUE
      CALL CROSS(EVEC,ZVEC,SVEC)
      DO 114 INN=1,3
         SVEC(INN)=SVEC(INN)/SVEC(0)
 114  CONTINUE

C     SET UP THE TRANSFORMATION FOR IJK = TRM*SEZ

      DO 120 INL=1,3
         TRM(INL,1)=SVEC(INL)
         TRM(INL,2)=EVEC(INL)
         TRM(INL,3)=ZVEC(INL)
 120  CONTINUE
      DO 121 INNL=1,3
         RE(INNL)=R(INNL)
         RSE(INNL)=RS(INNL)
 121  CONTINUE

C     CONVERT TO SEZ FOR CALCULATIONS

      DO 130 INK=1,3
         RHOVEC(INK)=RHOVE(1)*TRM(1,INK)+RHOVE(2)*TRM(2,INK)
     +               +RHOVE(3)*TRM(3,INK)
         R(INK)=RE(1)*TRM(1,INK)+RE(2)*TRM(2,INK)+
     +          RE(3)*TRM(3,INK)
         RS(INK)=RSE(1)*TRM(1,INK)+RSE(2)*TRM(2,INK)
     +           +RSE(3)*TRM(3,INK)
 130  CONTINUE

      IF (RHOVEC(1).EQ.0.0D+00) THEN
         IF (RHOVEC(2).GT.0.0D+00) AZ=90.0D+00*RAD
         IF (RHOVEC(2).LT.0.0D+00) AZ=270.0D+00*RAD
         IF (RHOVEC(2).EQ.0.0D+00) THEN
            AZ=0.0D+00
            IF (RHOVEC(3).GT.0.0D+00) EL=90.0D+00*RAD
            IF (RHOVEC(3).LT.0.0D+00) EL=-90.0D+00*RAD
         END IF
      END IF

      IF ((RHOVEC(1).NE.0.0D+00).AND.(RHOVEC(2).NE.0.0D+00)) THEN
         AZ=DATAN(RHOVEC(2)/RHOVEC(1))
         EL=DATAN(RHOVEC(3)/DSQRT(RHOVEC(1)*RHOVEC(1)+RHOVEC(2)*
     +      RHOVEC(2)))
         IF (RHOVEC(1).LT.0.0D+00) AZ=AZ+180.0D+00*RAD
         IF ((RHOVEC(1).GT.0.0D+00).AND.(RHOVEC(2).LT.0.0D+00)) AZ=
     +      AZ+360.0D+00*RAD
      END IF

      RETURN
      END

      SUBROUTINE MMPY(MAT1,MAT1R,MAT1C,MAT2,MAT2C,MAT3)
```

```
      DOUBLE PRECISION MAT1(MAT1R,MAT1C),MAT2(MAT1C,MAT2C),
     +       MAT3(MAT1R,MAT2C)

      INTEGER IMA,IMB,IMC,MAT1R,MAT1C,MAT2C

      DO 4000 IMA=1,MAT1R
         DO 4000 IMB=1,MAT2C
            MAT3(IMA,IMB)=0.0D+00
            DO 4000 IMC=1,MAT1C
               MAT3(IMA,IMB)=MAT3(IMA,IMB)+
     +                       MAT1(IMA,IMC)*MAT2(IMC,IMB)
 4000 CONTINUE

      RETURN
      END

      SUBROUTINE MTRANS(MAT4,MAT4R,MAT4C,MAT5)

      DOUBLE PRECISION MAT4(MAT4R,MAT4C),MAT5(MAT4C,MAT4R)

      INTEGER MAT4R,MAT4C

      INTEGER IMD,IME

      DO 4020 IMD=1,MAT4R
         DO 4020 IME=1,MAT4C
            MAT5(IME,IMD)=MAT4(IMD,IME)
 4020 CONTINUE

      RETURN
      END

      SUBROUTINE MATPRT(MAT6,MAT6R,MAT6C)

      DOUBLE PRECISION MAT6(MAT6R,MAT6C)

      INTEGER MAT6R,MAT6C

      INTEGER IMH,IMI

 4040 FORMAT(10(1X,E12.6))
      DO 4030 IMH=1,MAT6R
         WRITE(*,4040) (MAT6(IMH,IMI),IMI=1,MAT6C)
 4030 CONTINUE

      RETURN
      END

      SUBROUTINE OBSER(TOB,Q1,ZPRED,H,NXT,Z,RINTRK,NPTS,
     +                 ILCC,TIMTRK,SIGTRK)

      DOUBLE PRECISION TOB,Q1(3,3),ZPRED(3),H(3,8),Z(3),RINTRK(500),
```

```
     +                    TIMTRK(500),SIGTRK

      INTEGER NXT,ILCC,NPTS

      COMMON /HAM/ T,Y(72,4),F(72,4),ERR(72),N,DT,MODE,TEPOCH

      DOUBLE PRECISION T,Y,F,ERR,DT,TEPOCH

      INTEGER N,MODE

      DOUBLE PRECISION TO,R(0:3),V(0:3),RHO,AZ,EL,RS(0:3),SIGEL,
     +       OHM1,AZDNOM,ELDNOM,ELBTM,HIT(3,3),TRM(3,3),SIGRHO,
     +       SIGAZ,RO(3),ROBSER(3),RTRK(3),ROB(3),SIG1,SIG2,SIG3,
     +       VSEZ(0:3),SIGTOT

      INTEGER IOA,IOC,IOD,IOE,IOF,IOT,IOU,IOV,IOW,IOX,IOY,IOZ,IOZZ

C     RANGE - AZIMUTH - ELEVATION DATA

C     Q INVERSE MATRIX

      DO 10 IOC=1,3
         DO 10 IOD=1,3
            Q1(IOC,IOD)=0.0D+00
10    CONTINUE

C     SPECIFY SIGMA RHO, AZ, AND EL

      SIGRHO=.00001D+00
      SIGAZ=.001D+00
      SIGEL=.001D+00
      Q1(1,1)=SIGRHO*SIGRHO
      Q1(2,2)=SIGAZ*SIGAZ
      Q1(3,3)=SIGEL*SIGEL
      DO 17 IOE=1,3
         Q1(IOE,IOE)=1.0D+00/Q1(IOE,IOE)
17    CONTINUE
      TO=0.0D+00
      DO 11 IOA=1,3
         R(IOA)=Y(IOA,NXT)
         V(IOA)=Y(IOA+3,NXT)
11    CONTINUE

      CALL RAZEL(R,V,RHO,AZ,EL,TO,T,RS,TRM)

C     COMPUTE THE INTRACK RESIDUALS FOR FIRST PASS ONLY

      IF (ILCC.EQ.1) THEN

C     COMPUTE THE SIGMA FOR INTRACK RESIDUALS

C     FIRST WE NEED VELOCITY IN SEZ FRAME
```

```
              DO 12 IOF=1,3
                 VSEZ(IOF)=TRM(1,IOF)*V(1)+TRM(2,IOF)*V(2)+TRM(3,IOF)*
     +                 V(3)
12            CONTINUE

              CALL MAG(VSEZ)

C       COMPUTE THE COEFFICIENTS FOR INTRACK SIGMA

              SIG1=VSEZ(1)*DCOS(Z(2))*DCOS(Z(3))+VSEZ(2)*DSIN(Z(2))*
     +          DCOS(Z(3))+VSEZ(3)*DSIN(Z(3))

              SIG2=VSEZ(2)*Z(1)*DCOS(Z(2))*DCOS(Z(3))-VSEZ(1)*Z(1)*
     +          DSIN(Z(2))*DCOS(Z(3))

              SIG3=VSEZ(3)*Z(1)*DCOS(Z(3))-VSEZ(1)*Z(1)*DCOS(Z(2))*
     +          DSIN(Z(3))-VSEZ(2)*Z(1)*DSIN(Z(2))*DSIN(Z(3))

C       COMPUTE SIGMA INTRACK

              SIGTOT=DSQRT(SIG1*SIG1/Q1(1,1)+SIG2*SIG2/Q1(2,2)+
     +            SIG3*SIG3/Q1(3,3))/VSEZ(0)
              SIGTOT=SIGTOT*1.0D-01

C       NOW DETERMINE WORST SIGMA INTRACK FOR OBSERVATIONS

              IF(SIGTOT.GT.SIGTRK) SIGTRK=SIGTOT


C       CONVERT RHO OBSERVED TO SEZ FRAME COMPONENTS

              RO(1)=Z(1)*DCOS(Z(2))*DCOS(Z(3))
              RO(2)=Z(1)*DSIN(Z(2))*DCOS(Z(3))
              RO(3)=Z(1)*DSIN(Z(3))

C       R OF OBSERVATION IN SEZ FRAME EQUALS RHO IN SEZ FRAME
C       PLUS R OF RADAR SITE IN SEZ FRAME

              DO 13 IOX=1,3
                 ROB(IOX)=RO(IOX)+RS(IOX)
13            CONTINUE

C       MUST NOW CONVERT R OBSERVATION TO THE IJK FRAME

              DO 14 IOY=1,3
                 ROBSER(IOY)=TRM(IOY,1)*ROB(1)+TRM(IOY,2)*ROB(2)+
     +                   TRM(IOY,3)*ROB(3)
14            CONTINUE

C       RTRK = R OF OBSERVATION - R OF HAMING IN IJK FRAME

              DO 15 IOZ=1,3
                 RTRK(IOZ)=ROBSER(IOZ)-Y(IOZ,NXT)
```

```
15          CONTINUE

C        INTRACK RESIDUAL IS THEN COMPUTED AS RINTRK=RTRK DOT
C        VELOCITY VECTOR DIVIDED BY MAGNITUDE OF VELOCITY VECTOR

            RINTRK(NPTS)=0.0D+00
            CALL MAG(V)

            DO 16 IOZZ=1,3
               RINTRK(NPTS)=RINTRK(NPTS)+RTRK(IOZZ)*V(IOZZ)/V(0)
               TIMTRK(NPTS)=TOB
16          CONTINUE

         END IF

C        THIS CALCULATES THE G MATRIX

         ZPRED(1)=RHO
         ZPRED(2)=AZ
         ZPRED(3)=EL

C        THE H MATRIX

C        NOTE THAT R AND RS ARE IN SEZ

         OHM1=(R(1)-RS(1))*(R(1)-RS(1))+(R(2)-RS(2))*(R(2)-RS(2))
     +        +(R(3)-RS(3))*(R(3)-RS(3))

         H(1,1)=(1.0D+00/DSQRT(OHM1))*(R(1)-RS(1))
         H(1,2)=(1.0D+00/DSQRT(OHM1))*(R(2)-RS(2))
         H(1,3)=(1.0D+00/DSQRT(OHM1))*(R(3)-RS(3))
         H(1,4)=0.0D+00
         H(1,5)=0.0D+00
         H(1,6)=0.0D+00
         H(1,7)=0.0D+00
         H(1,8)=0.0D+00

         AZDNOM=1.0D+00+((R(2)-RS(2))/(R(1)-RS(1)))*
     +           ((R(2)-RS(2))/(R(1)-RS(1)))

         H(2,1)=(-(R(2)-RS(2))/((R(1)-RS(1))*(R(1)-RS(1))))/AZDNOM
         H(2,2)=(1.0D+00/(R(1)-RS(1)))/AZDNOM
         H(2,3)=0.0D+00
         H(2,4)=0.0D+00
         H(2,5)=0.0D+00
         H(2,6)=0.0D+00
         H(2,7)=0.0D+00
         H(2,8)=0.0D+00

         ELBTM=(R(1)-RS(1))*(R(1)-RS(1))+(R(2)-RS(2))*(R(2)-RS(2))
         ELDNOM=1.0D+00+((R(3)-RS(3))*(R(3)-RS(3)))/ELBTM

         H(3,1)=((-(R(1)-RS(1))*(R(3)-RS(3)))/DSQRT(ELBTM*ELBTM*ELBTM))
```

```
     +        /ELDNOM
      H(3,2)=((-(R(2)-RS(2))*(R(3)-RS(3)))/DSQRT(ELBTM*ELBTM*ELBTM))
     +        /ELDNOM
      H(3,3)=(1.0D+00/DSQRT(ELBTM))/ELDNOM
      H(3,4)=0.0D+00
      H(3,5)=0.0D+00
      H(3,6)=0.0D+00
      H(3,7)=0.0D+00
      H(3,8)=0.0D+00

C     CONVERT TO IJK FRAME

      DO 2010 IOT=1,3
         HIT(1,IOT)=H(1,IOT)
         HIT(2,IOT)=H(2,IOT)
         HIT(3,IOT)=H(3,IOT)
2010  CONTINUE
      DO 2020 IOU=1,3
         DO 2020 IOV=1,3
            H(IOU,IOV)=0.0D+00
            DO 2020 IOW=1,3
               H(IOU,IOV)=HIT(IOU,IOW)*TRM(IOV,IOW)+H(IOU,IOV)
2020  CONTINUE

      RETURN
      END

      SUBROUTINE STAGE(RINTRK,ILNOB,TIMTRK,SIGTRK,ISTAGE,XSTAGE,
     +                 XREFU,TSTAGE,PSTG)

      DOUBLE PRECISION RINTRK(500),TIMTRK(500),SIGTRK,XREFU(8,1),
     +        TSTAGE,XSTAGE(2),PSTG(2,2)

      INTEGER ILNOB,ISTAGE

      DOUBLE PRECISION TCOMM,TIMRES,QINV(1,1),
     +        ZSTG,TIMOBS,ZSTGPRD,ANOT,DELX(2,1),TTQIT(2,2),
     +        TTQIR(2,1),TMTSTG(1,2),TDIFF,TTQI(2,1),RESSTG(100),
     +        TMSTGT(2,1),PSTGI(2,2),WORK(2),TNTOLD

      INTEGER ITA,ITC,ITD,ITE,ITF,ITG,ITH,ITI,ITJ,ITK,ICNT,RCNT,
     +        DONE,ITL,ITM,NITS,SDONE,ITN

C     FIRST WE SEND THE SIGMA IN TRACK VALUE TO THE OUTPUT FILE

200   FORMAT (/,2X,'SIGMA IN-TRACK = ',E20.13,/)

      WRITE(18,200) SIGTRK

C     MUST NOW DETERMINE A STAGING EVENT HAS TAKEN PLACE AND WILL
C     KEEP TRACK OF THE OBSERVATION THE STAGING EVENT SINGLED OUT

      DONE=0
```

```
      ISTAGE=0
      RCNT=0
      ICNT=0

      DO 21 ITA = 1,ILNOB
         RCNT=RCNT+1
         IF (ABS(RINTRK(ITA)).GT.(3.0D+00*SIGTRK)) THEN
            ICNT=ICNT+1
            GOTO 20
         END IF
         ICNT=0
20       IF (ICNT.EQ.3) THEN
            ISTAGE=1
            GOTO 22
         END IF
21    CONTINUE

22    CONTINUE

      RCNT =RCNT-3

C     NOW WE BEGIN THE ESTIMATION OF THE PARAMETERS OF THE NEXT
C     STAGE ONCE A STAGING EVENT HAS BEEN DETECTED BY USING THE
C     REMAINING RESIDUALS

      IF (ISTAGE.EQ.1) THEN

C     FILL PLOT OF THE STAGING EVENT RESIDUALS

201      FORMAT (2(2X,E20.13))
         DO 23 ITL=1,ILNOB
            WRITE (19,201) RINTRK(ITL),TIMTRK(ITL)
23       CONTINUE

         PRINT*,'A STAGING EVENT HAS BEEN DETECTED'

C     WILL NOW PRINT OUT THE LAST VALUE FOR THE MAIN STATE VECTOR

230      FORMAT (/,2X,'LAST GOOD VALUES FOR THE MAIN STATE VECTOR',
     +            /,/,4(2X,E18.11),/,4(2X,E18.11),/)
         WRITE (18,230) XREFU

C     FIRST WE MUST INPUT OUR INITIAL GUESS FOR THE STAGING
C     STATE VECTOR; VE*M AND TSTAGE

         PRINT*,'INPUT INITIAL GUESS FOR THE STAGING STATE VECTOR'
         READ*,XSTAGE(1),XSTAGE(2)

         PRINT*,'INPUT THE MAX LEAST SQUARES ITERATIONS TO RUN'
         READ*,NITS

C     PRINT OUT THE HEADER FOR THE STAGING ESTIMATOR
```

```
202         FORMAT (/,27X,'NONLINEAR LS STAGING ESTIMATOR',/,/,2X,
     +             'INITIAL GUESS IS :',/,/,2X,
     +             'VE*M = ',E18.11,2X,'TSTAGE = ',
     +             E18.11)
            WRITE (18,202) XSTAGE(1),XSTAGE(2)

C       MUST THEN INITIALIZE T TRANSPOSE * Q INVERSE * T AND
C       T TRANSPOSE * Q INVERSE * RESIDUAL SUMATIONS AND SPECIFY
C       THE VALUE OF Q INVERSE WHERE Q INVERSE IS A 1 BY 1 MATRIX
C       CONSISTING OF THE RECIPROCAL OF THE SQUARE OF THE IN TRACK
C       SIGMA AS USED PREVIOUSLY AS SIGTRK

            QINV(1,1)=1.0D+00/(SIGTRK*SIGTRK)

C       WILL NOW PRINT OUT THE VALUE OF Q INVERSE

231         FORMAT (/,2X,'Q INVERSE, A 1 X 1 MATRIX, IS :',/,/,2X,
     +              E18.11)
            WRITE (18,231) QINV(1,1)

            DO 25 ITE=1,2
              TTQIR(ITE,1)=0.0D+00
              DO 25 ITL=1,2
                TTQIT(ITE,ITL)=0.0D+00
25          CONTINUE

C       WILL NOW START THE NONLINEAR LEAST SQUARES ITERATION LOOP

            DO 100 ITD=1,NITS

C       PRINT FIRST OR LAST PASS RESIDUAL HEADERS WHEN NECESSARY

203           FORMAT (/,2X,'FIRST PASS RESIDUALS : ',/)
204           FORMAT (/,2X,'LAST PASS RESIDUALS :',/)
              IF (ITD.EQ.1) WRITE (18,203)
              IF (DONE.EQ.1) WRITE (18,204)

C       NOW IT WILL PROCESS THE OBSERVATIONS

              DO 99 ITF=RCNT,ILNOB

C       READ IN THE OBSERVATIONS

                ZSTG=RINTRK(ITF)
                TIMOBS=TIMTRK(ITF)

C       NOW COMPUTE THE PREDICTED VALUE OF ZSTG (ZSTGPRD)

                TIMRES=XSTAGE(2)-TSTAGE
                TNTOLD=1.0D+00-XREFU(8,1)*TIMRES
                TDIFF=TIMOBS-XSTAGE(2)

                ANOT=(XREFU(7,1)*XREFU(8,1)/TNTOLD)-XSTAGE(1)
```

```
                 ZSTGPRD=-ANOT*TDIFF*TDIFF/2.0D+00

C        NOW COMPUTE RESIDUALS FOR EACH OBSERVATION

                 RESSTG(ITF)=ZSTG-ZSTGPRD

C        PRINT OUT FIRST FIVE RESIDUAL VALUES FOR FIRST AND LAST PASS

                 IF (ITF.LT.(RCNT+5)) GOTO 30
                 IF ((DONE.EQ.1).AND.(ITF.LT.(RCNT+5))) GOTO 30
                 IF ((DONE.EQ.1).AND.(ITF.GE.(RCNT+5))) GOTO 35
                 GOTO 40

30               CONTINUE

220              FORMAT (/,2X,'RESIDUAL = ',E18.11)
                 WRITE (18,220) RESSTG(ITF)

35               CONTINUE
                 IF (DONE.EQ.1) GOTO 55

40               CONTINUE

C        CALCULATE T MATRIX FOR EACH OBSERVATION

                 TCOMM=XREFU(7,1)*XREFU(8,1)*XREFU(8,1)*TDIFF*
     +                TDIFF/(TNTOLD*TNTOLD)

                 TMTSTG(1,1)=TDIFF*TDIFF/2.0D+00

                 TMTSTG(1,2)=-TCOMM/2.0D+00+ANOT*TDIFF

C        NOW THAT THE T MATRIX HAS BEEN CALCULATED WE CAN SUM
C        TTQIT AND TTQIR.  FIRST WE WILL COMPUTE T TRANSPOSE *
C        Q INVERSE

                 CALL MTRANS(TMTSTG,1,2,TMSTGT)
                 CALL MMPY(TMSTGT,2,1,QINV,1,TTQI)

C        NOW WE FORM THE SUMATION T TRANSPOSE * Q INVERSE * T

                 DO 45 ITG=1,2
                    DO 45 ITH=1,2
                       TTQIT(ITG,ITH)=TTQIT(ITG,ITH)+TTQI(ITG,1)*
     +                               TMTSTG(1,ITH)
45               CONTINUE

C         NOW WE FORM THE SUMATION T TRANSPOSE * Q INVERSE * RESIDUAL

                 DO 50 ITI=1,2
                    TTQIR(ITI,1)=TTQIR(ITI,1)+TTQI(ITI,1)*RESSTG(ITF)
50               CONTINUE
```

```
55            CONTINUE

99          CONTINUE

C       DID WE JUST PRINT THE LAST PASS RESIDUALS ?

            IF (DONE.EQ.1) GOTO 110

C       ONCE ALL OBSERVATIONS HAVE BEEN PROCESSED
C       MUST NOW FIND THE COVARIENCE MATRIX P BY FINDING THE INVERSE
C       OF THE SUMMATION OF T TRANSPOSE * Q INVERSE * T

            CALL MEQL(TTQIT,2,2,PSTGI)

236     FORMAT (/,2X,'P INVERSE MATRIX IS :',/,/,2(2X,E18.11),/,
     +       2(2X,E18.11))
        WRITE (18,236) PSTGI

            CALL LINV1F(PSTGI,2,2,PSTG,0,WORK,IER)

235         FORMAT (/,2X,'P MATRIX IS :',/,/,2(2X,E18.11),/,
     +       2(2X,E18.11))
            WRITE (18,235) PSTG

C       NOW COMPUTE DX = P * T TRANSPOSE *Q INVERSE * RESIDUAL

            CALL MMPY(PSTG,2,2,TTQIR,1,DELX)

C       NOW UPDATE THE GUESS

            DO 60 ITJ=1,2
               XSTAGE(ITJ)=XSTAGE(ITJ)+DELX(ITJ,1)
60          CONTINUE

C       PRINT ITERATION NUMBER AND CURRENT STATE VECTOR

205         FORMAT (/,2X,'ITERATION ',I3,/,/,2X,'STATE CORRECTIONS',
     +              /,2(2X,E18.11))
            WRITE (18,205) ITD,DELX(1,1),DELX(2,1)

206         FORMAT (/,2X,'CURRENT STAGING STATE VECTOR ',/,2X,
     +              'VE*M = ',E18.11,2X,'TSTAGE = ',E18.11,/)
            WRITE (18,206)XSTAGE

C       NOW WE CAN CHECK FOR CONVERGENCE

            SDONE=0
            DO 65 ITK=1,2
               IF (DABS(DELX(ITK,1)).GT.0.01*DSQRT(PSTG(ITK,ITK)))
     +            SDONE=1
65          CONTINUE
```

```
            IF (SDONE.EQ.0) DONE=1

C       REINITIALIZE T TRANSPOSE * Q INVERSE * T AND T TRANSPOSE
C       * Q INVERSE * RESIDUAL SUMMATIONS BEFORE NEXT ITERATION

            DO 70 ITM=1,2
               TTQIR(ITM,1)=0.0D+00
               DO 70 ITN=1,2
                  TTQIT(ITM,ITN)=0.0D+00
70          CONTINUE

C       WILL NOW UPDATE THE OBSERVATION COUNTER (RCNT) TO INCLUED
C       POINTS FROM THE ESTIMATED STAGING EVENT TIME TO THE END
C       OF THE AVAILABLE DATA (ILNOB)

            RCNT=0
            DO 75 ITN=1,ILNOB
               IF (XSTAGE(2).LT.TIMTRK(ITN)) GOTO 80
               RCNT=RCNT+1
75          CONTINUE

80          CONTINUE

100       CONTINUE

C       FAIURE FOR LEAST SQUARES

207       FORMAT (/,2X,' MAX ITERATION LIMIT HAS BEEN EXCEEDED ',
     +          'WITHOUT CONVERGENCE')
          WRITE (18,207)
          PRINT 207
          GOTO 150

110       CONTINUE

208       FORMAT (/,2X,'CONVERGENCE HAS BEEN ACHEIVED',/,2X,
     +          'COVARIENCE MATRIX : '/,/,2(2X,E18.11),
     +          /,2(2X,E18.11))
          WRITE (18,208)PSTG
          PRINT*,'CONVERGENCE HAS BEEN ACHEIVED'

        END IF

150     CONTINUE

        RETURN
        END

$
```

APPENDIX F

COMPUTER OUTPUTS

Once the Bayes Filter estimation routine was functioning as it was designed to, test cases involving both space based sensor data and land based sensor data were processed. The following two test runs are an indication as to how well the the algorithm was able to detect the staging event and its subsequent estimation of launch vehicle parameters for stages one and two. The first case involved a space based sensor with the following inputs.

Vehicle Type: Titan 34 D

Time of Flight: 800.0 Seconds

Epoch Time: .0020 TUs

Launch Site: 53.7 Degrees North

158.2 Degrees East

Elevation: 1.0 DU

Sensor type: Space based

Semi-major Axis	2.5 DUs
Eccentricity	.25
Inclination	45.0 Degrees
Longitude Of Ascending Node	10.0 Degrees
Argument of Periapsis	10.0 Degrees

Sensor type: Land Based

52.6 Degrees North
174.1 Degrees East

Noisy Data: NO

As the output data is lengthy, shown here is only a portion

of the total output. It consists of the first few segments of data, then the staging estimator output, and finally the first few segments after staging.

Space Based Sensor

NONLINEAR BAYES FILTER

```
 INITIAL STATE VECTOR :
 -0.13260000000E+00-0.57700000000E+00 0.80590000000E+00-0.10230000000E-02
 -0.44490000000E-02 0.62640000000E-02 0.37380000000E+00 0.34660000000E+01
 INITIAL TIME : 0.002000 # OF DATA POINTS :     0
 MAX LS ITERATIONS :          20  # OF BAYES CHUNKS :    50
 MAX BAYES ITERATIONS  :      20  RANK OF P  :           8
 BETA MATRIX =  1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000

 FIRST PASS RESIDUALS:

TIME, RES =  2.4957785237078558E-03  1.2471533923696931E-05
-4.1606240044511633E-05 -1.2652892461550991E-07
TIME, RES =  2.9915570474157117E-03  1.2473467657536652E-05
-4.1610263675051495E-05 -1.2434843579822719E-07
TIME, RES =  3.4873355711235675E-03  1.2475337674511255E-05
-4.1614616825447204E-05 -1.2214112704578284E-07
TIME, RES =  3.9831140948314234E-03  1.2477139197109022E-05
-4.1619320691244077E-05 -1.1990566792241530E-07

 ITERATION   1

 STATE CORRECTIONS
 -0.11309339093E-04 0.30464525995E-04 0.28282641496E-04 0.40945949387E-06
 -0.13957263612E-06 0.12810035276E-04 0.76595344581E-04-0.54924155120E-03

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13261130934E+00-0.57696953547E+00 0.80592828264E+00-0.10225905405E-02
 -0.44491395726E-02 0.62768100353E-02 0.37387659534E+00 0.34654507584E+01

TIME, RES =  2.4957785237078558E-03 -3.9192687983913288E-10
-8.2518381017138154E-11 -1.4480680543549340E-10
TIME, RES =  2.9915570474157117E-03 -3.8782438371853800E-10
-9.1681717773184346E-11 -1.4276252402467549E-10
TIME, RES =  3.4873355711235675E-03 -3.8462921736481803E-10
-1.0159606489423822E-10 -1.4138007431441224E-10
TIME, RES =  3.9831140948314234E-03 -3.8229130971956238E-10
-1.1231049423798822E-10 -1.4062945252746317E-10

 ITERATION   2

 STATE CORRECTIONS
  0.20971850799E-09 0.33441979242E-09-0.34102483818E-09-0.38685863784E-08
  0.30343688610E-08 0.23657136994E-07-0.59597386603E-04 0.52813333676E-03

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13261130913E+00-0.57696953514E+00 0.80592828230E+00-0.10225944091E-02
 -0.44491365383E-02 0.62768336924E-02 0.37381699796E+00 0.34659788918E+01
```

LAST PASS RESIDUALS:

```
TIME, RES =  2.4957785237078558E-03  9.5645713571457236E-14
-2.2204460492503131E-15  6.7890137955828322E-14
TIME, RES =  2.9915570474157117E-03  3.8402614421784165E-13
-4.7184478546569153E-15  2.5957014315736160E-13
TIME, RES =  3.4873355711235675E-03  8.6525231424161575E-13
-8.8262730457699945E-15  5.7973070788364112E-13
TIME, RES =  3.9831140948314234E-03  1.5406009801210985E-12
-1.4876988529977098E-14  1.0296763441886014E-12
```

CONVERGENCE ACHIEVED.
IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
DECLARIUM EST ESTIMATIA

COVARIANCE MATRIX AT EPOCH IS:

```
 0.3903003E-07-0.4647810E-07 0.6463529E-07-0.3220947E-06 0.6177572E-06
-0.8604711E-06 0.2759423E-02-0.2541794E-01

-0.4647810E-07 0.8326810E-07-0.5708883E-07 0.3861013E-06-0.1028979E-05
 0.8447621E-06-0.3286934E-02 0.3027752E-01

 0.6463529E-07-0.5708883E-07 0.1214895E-06-0.5307781E-06 0.8547547E-06
-0.1613920E-05 0.4669196E-02-0.4298711E-01

-0.3220947E-06 0.3861013E-06-0.5307781E-06 0.5565299E-05-0.9360608E-05
 0.1294712E-04-0.4411564E-01 0.4071111E+00

 0.6177572E-06-0.1028979E-05 0.8547547E-06-0.9360608E-05 0.2902532E-04
-0.2977864E-04 0.9511079E-01-0.8728340E+00

-0.8604711E-06 0.8447621E-06-0.1613920E-05 0.1294712E-04-0.2977864E-04
 0.4999116E-04-0.1348810E+00 0.1237383E+01

 0.2759423E-02-0.3286934E-02 0.4669196E-02-0.4411564E-01 0.9511079E-01
-0.1348810E+00 0.4107946E+03-0.3776804E+04

-0.2541794E-01 0.3027752E-01-0.4298711E-01 0.4071111E+00-0.8728340E+00
 0.1237383E+01-0.3776804E+04 0.3472634E+05
```

BEGIN NEXT BAYES LOOP

FIRST PASS RESIDUALS:

```
TIME, RES =  2.7284704709100647E-02  2.6514423989709712E-10
-2.4041879598257765E-13  1.7619125602941210E-10
TIME, RES =  2.7780483232808503E-02  2.7602270469273549E-10
-4.2021941482062175E-14  1.8338669471873459E-10
TIME, RES =  2.8276261756516359E-02  2.8713670330304808E-10
 1.8635093468333253E-13  1.9073448376261126E-10
TIME, RES =  2.8772040280224215E-02  2.9848823412947922E-10
```

```
4.4048098502003086E-13  1.9823623298442783E-10

SIGMA IN-TRACK =  0.1876563478352E-03


ITERATION   1

STATE CORRECTIONS
-0.60443228642E-10-0.26295044084E-09 0.36800864849E-09-0.50937188823E-08
-0.22163236041E-07 0.30398170500E-07 0.88831160205E-08 0.36375652277E-05

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13264929669E+00-0.57713481264E+00 0.80616992864E+00-0.20694915263E-02
-0.90040100775E-02 0.13751501020E-01 0.37381700684E+00 0.34659825294E+01


LAST PASS RESIDUALS:

TIME, RES =  2.7284704709100647E-02  3.3306690738754696E-16
-6.1062266354383610E-16  2.3869795029440866E-15
TIME, RES =  2.7780483232808503E-02  3.2196467714129540E-15
-2.2759572004815709E-15  3.8857805861880479E-15
TIME, RES =  2.8276261756516359E-02  6.8833827526759706E-15
-5.5511151231257827E-16  5.9119376061289586E-15
TIME, RES =  2.8772040280224215E-02  1.2101430968414206E-14
-1.7763568394002505E-15  9.5201624361607173E-15

CONVERGENCE ACHIEVED.
IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
DECLARIUM EST ESTIMATIA


COVARIANCE MATRIX AT EPOCH IS:
 0.5326076E-07-0.6281594E-07 0.8771866E-07-0.1283216E-05 0.1548222E-05
-0.2165235E-05 0.1071689E-02-0.9441056E-02

-0.6281594E-07 0.1093351E-06-0.7830126E-07 0.1513152E-05-0.2685997E-05
 0.1954115E-05-0.1202959E-02 0.1061479E-01

 0.8771866E-07-0.7830126E-07 0.1627011E-06-0.2109102E-05 0.1964187E-05
-0.4044935E-05 0.1850221E-02-0.1625326E-01

-0.1283216E-05 0.1513152E-05-0.2109102E-05 0.4921990E-04-0.5840715E-04
 0.8144504E-04-0.3963665E-01 0.3502494E+00

 0.1548222E-05-0.2685997E-05 0.1964187E-05-0.5840715E-04 0.1076256E-03
-0.8236889E-04 0.5277039E-01-0.4608560E+00

-0.2165235E-05 0.1954115E-05-0.4044935E-05 0.8144504E-04-0.8236889E-04
 0.1666931E-03-0.8121947E-01 0.7064677E+00

 0.1071689E-02-0.1202959E-02 0.1850221E-02-0.3963665E-01 0.5277039E-01
-0.8121947E-01 0.4644291E+02-0.3995065E+03
```

```
-0.9441056E-02 0.1061479E-01-0.1625326E-01 0.3502494E+00-0.4608560E+00
 0.7064677E+00-0.3995065E+03 0.3442251E+04


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  5.2073630894493458E-02  2.1186941090434175E-12
-2.0283774659901610E-13  1.4370171719235714E-12
TIME, RES =  5.2569409418201314E-02  2.2054580384178735E-12
-2.1016521856154213E-13  1.4958867478043203E-12
TIME, RES =  5.3065187941909170E-02  2.2940538357829610E-12
-2.1938006966593093E-13  1.5562828803439288E-12
TIME, RES =  5.3560966465617026E-02  2.3845370122899112E-12
-2.2776225350185086E-13  1.6183998585717063E-12

 SIGMA IN-TRACK =  0.1877475851419E-03


 ITERATION   1

 STATE CORRECTIONS
 -0.45514575682E-12-0.19782258736E-11 0.30948135153E-11-0.37428283587E-10
 -0.16583823041E-09 0.25908692960E-09 0.11485310663E-09 0.24220206542E-07

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13271389423E+00-0.57741586570E+00 0.80664307132E+00-0.31096134908E-02
 -0.13529406065E-01 0.25228387479E-01 0.37381700696E+00 0.34659825536E+01


 LAST PASS RESIDUALS:

TIME, RES =  5.2073630894493458E-02  2.2204460492503131E-16
-4.4408920985006262E-16  9.4368957093138306E-16
TIME, RES =  5.2569409418201314E-02  1.1102230246251565E-16
 3.3306690738754696E-16  6.3837823915946501E-16
TIME, RES =  5.3065187941909170E-02 -1.6653345369377348E-16
-6.6613381477509392E-16  6.1062266354383610E-16
TIME, RES =  5.3560966465617026E-02 -3.8857805861880479E-16
-6.6613381477509392E-16  9.7144514654701197E-16

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.6072234E-07-0.7147296E-07 0.9986768E-07-0.2061610E-05 0.2424570E-05
-0.3386811E-05 0.3580830E-03-0.3118665E-02

-0.7147296E-07 0.1242337E-06-0.8890348E-07 0.2424957E-05-0.4221118E-05
```

```
 0.3025866E-05-0.3686811E-03 0.3244705E-02

 0.9986768E-07-0.8890348E-07 0.1849888E-06-0.3386482E-05 0.3033242E-05
-0.6309290E-05 0.6606039E-03-0.5677114E-02

-0.2061610E-05 0.2424957E-05-0.3386482E-05 0.1038010E-03-0.1212931E-03
 0.1690747E-03-0.1704629E-01 0.1498992E+00

 0.2424570E-05-0.4221118E-05 0.3033242E-05-0.1212931E-03 0.2145960E-03
-0.1571304E-03 0.2267489E-01-0.1928854E+00

-0.3386811E-05 0.3025866E-05-0.6309290E-05 0.1690747E-03-0.1571304E-03
 0.3268897E-03-0.4113979E-01 0.3424153E+00

 0.3580830E-03-0.3686811E-03 0.6606039E-03-0.1704629E-01 0.2267489E-01
-0.4113979E-01 0.1185081E+02-0.9033076E+02

-0.3118665E-02 0.3244705E-02-0.5677114E-02 0.1498992E+00-0.1928854E+00
 0.3424153E+00-0.9033076E+02 0.6936491E+03


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  7.6862557079886247E-02  1.6542323066914832E-14
-6.8278716014447127E-15  1.2517764602648640E-14
TIME, RES =  7.7358335603594102E-02  1.7319479184152442E-14
-6.8833827526759706E-15  1.3600232051658168E-14
TIME, RES =  7.7854114127301957E-02  1.7874590696465020E-14
-7.7715611723760958E-15  1.4516166046973922E-14
TIME, RES =  7.8349892651009813E-02  1.8984813721090177E-14
-8.3266726846886741E-15  1.4349632593280148E-14

 SIGMA IN-TRACK =  0.1871505838125E-03


 ITERATION   1

 STATE CORRECTIONS
 -0.28785158617E-14-0.12297194010E-13 0.27942401834E-13 0.19573092864E-12
 -0.14096522014E-11 0.31650196992E-11-0.69360935084E-10 0.69987391100E-09

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13280006717E+00-0.57779078968E+00 0.80746713138E+00-0.37386480479E-02
 -0.16266229781E-01 0.42310351301E-01 0.37381700689E+00 0.34659825543E+01


 LAST PASS RESIDUALS:

TIME, RES =  7.6862557079886247E-02  5.5511151231257827E-17
-1.6653345369377348E-16  4.4408920985006262E-16
TIME, RES =  7.7358335603594102E-02  1.1102230246251565E-16
```

```
 1.6653345369377348E-16  8.3266726846886741E-16
TIME, RES =  7.7854114127301957E-02  0.0000000000000000E+00
-4.4408920985006262E-16  9.7144514654701197E-16
TIME, RES =  7.8349892651009813E-02  3.8857805861880479E-16
-5.5511151231257827E-16  8.3266726846886741E-17

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.6564928E-07-0.7715554E-07 0.1078736E-06-0.2641064E-05 0.3100390E-05
-0.4331122E-05 0.1026425E-03-0.9849005E-03

-0.7715554E-07 0.1340107E-06-0.9580887E-07 0.3102206E-05-0.5391204E-05
 0.3857853E-05-0.8962399E-04 0.9116841E-03

 0.1078736E-06-0.9580887E-07 0.1996734E-06-0.4336550E-05 0.3862641E-05
-0.8056665E-05 0.2198114E-03-0.1978954E-02

-0.2641064E-05 0.3102206E-05-0.4336550E-05 0.1521423E-03-0.1780296E-03
 0.2482631E-03-0.5264764E-02 0.5256244E-01

 0.3100390E-05-0.5391204E-05 0.3862641E-05-0.1780296E-03 0.3116213E-03
-0.2261788E-03 0.7947383E-02-0.7029946E-01

-0.4331122E-05 0.3857853E-05-0.8056665E-05 0.2482631E-03-0.2261788E-03
 0.4752676E-03-0.2021447E-01 0.1622613E+00

 0.1026425E-03-0.8962399E-04 0.2198114E-03-0.5264764E-02 0.7947383E-02
-0.2021447E-01 0.6560231E+01-0.4295668E+02

-0.9849005E-03 0.9116841E-03-0.1978954E-02 0.5256244E-01-0.7029946E-01
 0.1622613E+00-0.4295668E+02 0.2835853E+03
```

```
SIGMA IN-TRACK =  0.1617379723854E-03


LAST GOOD VALUES FOR THE MAIN STATE VECTOR

-0.13273855504E+00  -0.57752316071E+00   0.82341630058E+00   0.10999752528E-01
 0.47858075971E-01   0.26249364632E+00   0.37381700697E+00   0.34659825541E+01

                        NONLINEAR LS STAGING ESTIMATOR

INITIAL GUESS IS :

VE*M =  0.98000000000E+00  TSTAGE =  0.20450000000E+00

Q INVERSE, A 1 X 1 MATRIX, IS :

 0.38227510069E+08

FIRST PASS RESIDUALS :


RESIDUAL = -0.34908531199E-04

RESIDUAL = -0.38484645040E-04

RESIDUAL = -0.42303032656E-04

RESIDUAL = -0.46372817240E-04

RESIDUAL = -0.50703258280E-04

P INVERSE MATRIX IS :

 0.11274575098E+02   0.34898810099E+04
 0.34898810099E+04   0.10886556751E+07

P MATRIX IS :

 0.11476361360E+02  -0.36789534553E-01
-0.36789534553E-01   0.11885401506E-03

ITERATION   1

STATE CORRECTIONS
```

```
-0.64785121011E+00   0.10489440517E-02

CURRENT STAGING STATE VECTOR
VE*M =  0.33214878989E+00  TSTAGE =  0.20554894405E+00


RESIDUAL = -0.96466498137E-06

RESIDUAL = -0.22165058416E-05

RESIDUAL = -0.41530309191E-05

RESIDUAL = -0.54014984629E-05

RESIDUAL = -0.67672831894E-05

P INVERSE MATRIX IS :

 0.11724070832E+02   0.54152808843E+04
 0.54152808843E+04   0.27073976235E+07

P MATRIX IS :

 0.11204008500E+01  -0.22410026710E-02
-0.22410026710E-02   0.48517657000E-05

ITERATION   2

STATE CORRECTIONS
 0.16147025867E+00  -0.55865780433E-03

CURRENT STAGING STATE VECTOR
VE*M =  0.49361904856E+00  TSTAGE =  0.20499028625E+00


RESIDUAL =  0.91174165950E-07

RESIDUAL = -0.97301836928E-06

RESIDUAL = -0.11480209913E-05

RESIDUAL = -0.20544599332E-05

RESIDUAL = -0.23195934440E-05

P INVERSE MATRIX IS :

 0.13474499585E+02   0.57280692571E+04
 0.57280692571E+04   0.26386050221E+07

P MATRIX IS :

 0.96190454109E+00  -0.20881699928E-02
```

```
-0.20881699928E-02   0.49121343403E-05

ITERATION   3

STATE CORRECTIONS
-0.57025938159E-02  -0.43268247548E-05

CURRENT STAGING STATE VECTOR
VE*M =  0.48791645474E+00  TSTAGE =  0.20498595942E+00


LAST PASS RESIDUALS :


RESIDUAL =  0.87696189093E-07

RESIDUAL = -0.96808340799E-06

RESIDUAL = -0.11333257606E-05

RESIDUAL = -0.20286571019E-05

RESIDUAL = -0.22813356807E-05

CONVERGENCE HAS BEEN ACHEIVED
COVARIENCE MATRIX :

 0.96190454109E+00  -0.20881699928E-02
-0.20881699928E-02   0.49121343403E-05

                        BEGIN ESTIMATION OF THE NEXT STAGE

                        STAGING OCCURED AT : 165.3855  SECONDS


 AT THE START OF THE NEXT STAGE THE

COVARIANCE MATRIX AT EPOCH IS:
 0.4565959E-07-0.5268802E-07 0.7443739E-07-0.2753388E-05 0.3148689E-05
-0.4606050E-05-0.9374819E-05 0.6697224E-05

-0.5268802E-07 0.9109443E-07-0.6435272E-07 0.3178810E-05-0.5453415E-05
 0.4053552E-05 0.3108699E-05 0.8351015E-05

 0.7443739E-07-0.6435272E-07 0.1372751E-06-0.4465468E-05 0.3891761E-05
-0.8051657E-05-0.4561439E-04 0.1035079E-03

-0.2753388E-05 0.3178810E-05-0.4465468E-05 0.2983318E-03-0.3381989E-03
 0.5072766E-03 0.3611681E-03 0.1397353E-02

 0.3148689E-05-0.5453415E-05 0.3891761E-05-0.3381989E-03 0.5910273E-03
-0.3924988E-03-0.3158587E-02 0.8328221E-02
```

```
-0.4606050E-05 0.4053552E-05-0.8051657E-05 0.5072766E-03-0.3924988E-03
 0.1144375E-02-0.1310467E-01 0.4760086E-01

-0.9374819E-05 0.3108699E-05-0.4561439E-04 0.3611681E-03-0.3158587E-02
-0.1310467E-01 0.2204885E+01-0.3579046E+01

 0.6697224E-05 0.8351015E-05 0.1035079E-03 0.1397353E-02 0.8328221E-02
 0.4760086E-01-0.3579046E+01 0.2032392E+02


 INITIAL STATE VECTOR IS:

 -0.13268644573E+00-0.57729644189E+00 0.82462090473E+00 0.12371639351E-01
  0.53826925144E-01 0.27759580461E+00 0.37381700697E+00 0.13052280812E+01

 FIRST PASS RESIDUALS:

TIME, RES =  0.2052691947202211      -1.7598803359053505E-07
 4.6135639925060090E-07 -1.8454662573108571E-07
TIME, RES =  0.2057649732439289      -1.0882111289856056E-07
 2.8503683480174402E-07 -1.1407825081799494E-07
TIME, RES =  0.2062607517676368      -2.3800784788602264E-07
 6.2183238963564591E-07 -2.4927168426613910E-07
TIME, RES =  0.2067565302913446      -1.8806796192594177E-07
 4.8889777853888816E-07 -1.9660017203104729E-07

 ITERATION   1

 STATE CORRECTIONS
  0.20038276788E-05 0.33939718523E-05 0.80176544096E-05-0.88864796533E-04
 -0.37381172947E-03-0.12193851790E-02-0.44828781693E+00 0.22984568451E+01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268444191E+00-0.57729304791E+00 0.82462892238E+00 0.12282774554E-01
  0.53453113414E-01 0.27637641943E+00-0.74470809967E-01 0.36036849263E+01

TIME, RES =  0.2052691947202211      -2.9219953961989731E-07
 7.2263151770779466E-06 -2.2355961721565176E-06
TIME, RES =  0.2057649732439289      -1.5038931544530953E-08
 6.3548217317799249E-06 -1.9221591553431061E-06
TIME, RES =  0.2062607517676368       1.1629122553813431E-07
 5.8636362288977395E-06 -1.7613384180326097E-06
TIME, RES =  0.2067565302913446       4.7724226143186499E-07
 4.7696948324293942E-06 -1.3595958135159680E-06

 ITERATION   2

 STATE CORRECTIONS
 -0.27571934039E-06 0.30503223808E-04 0.18352567639E-04-0.16675382008E-03
 -0.12943785286E-02-0.66780748179E-03 0.20212795672E+00-0.35642334244E+01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
```

```
 -0.13268471763E+00-0.57726254469E+00 0.82464727495E+00 0.12116020734E-01
  0.52158734885E-01 0.27570861194E+00 0.12765714676E+00 0.39451501963E-01

TIME, RES =  0.2052691947202211      4.7391326235546316E-07
 4.1821416475573692E-05 -1.2468718335323370E-06
TIME, RES =  0.2057649732439289      6.0427764392301242E-07
 4.0308031349856055E-05 -9.8531632564125005E-07
TIME, RES =  0.2062607517676368      5.7059496111344643E-07
 3.9223181106773364E-05 -8.9550706444674333E-07
TIME, RES =  0.2067565302913446      7.4831656843299399E-07
 3.7583915060535045E-05 -5.8392349766567264E-07

 ITERATION   3

 STATE CORRECTIONS
  0.23015725778E-04-0.37869792058E-04 0.43905035797E-04-0.24302742204E-03
  0.58437473522E-03-0.30457235393E-02 0.57528015299E+00 0.87214084988E+01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13266170190E+00-0.57730041448E+00 0.82469117999E+00 0.11872993312E-01
  0.52743109621E-01 0.27266288840E+00 0.70293729974E+00 0.87608600007E+01

TIME, RES =  0.2052691947202211     -4.5414986651337585E-06
 3.5884931160190536E-05 -2.8419918019273460E-05
TIME, RES =  0.2057649732439289     -4.5110060472031144E-06
 3.5363895264040046E-05 -2.8222524700083351E-05
TIME, RES =  0.2062607517676368     -5.0585473813646864E-06
 3.6359070117175651E-05 -2.8630326952577834E-05
TIME, RES =  0.2067565302913446     -5.8097805028523020E-06
 3.7892908763326183E-05 -2.9251347657577309E-05

 ITERATION   4

 STATE CORRECTIONS
 -0.47376315002E-05 0.26050625553E-03 0.16457167369E-03-0.90223993991E-03
 -0.69422899610E-02-0.54535944618E-02-0.63457005756E+00 0.86117518823E-01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13266643953E+00-0.57703990823E+00 0.82485575166E+00 0.10970753372E-01
  0.45800819660E-01 0.26720929394E+00 0.68367242181E-01 0.88469775195E+01

TIME, RES =  0.2052691947202211     -2.0504869135962167E-06
 3.3682774033860774E-04 -2.1425934319613260E-05
TIME, RES =  0.2057649732439289     -1.7702387292706234E-06
 3.3047293405286116E-04 -2.0470209393913530E-05
TIME, RES =  0.2062607517676368     -1.6952127908087533E-06
 3.2465169305578856E-04 -1.9728983019801083E-05
TIME, RES =  0.2067565302913446     -1.4500715173060286E-06
 3.1838159851843573E-04 -1.8808890836885528E-05

 ITERATION   5

 STATE CORRECTIONS
```

```
-0.40834119701E-04-0.24624169460E-03-0.28607695188E-03 0.17593223190E-02
 0.83301264799E-02 0.12693161187E-01 0.45668740767E-01-0.99248141841E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13270727365E+00-0.57728614992E+00 0.82456967471E+00 0.12730075691E-01
 0.54130946140E-01 0.27990245513E+00 0.11403598295E+00 0.78544961011E+01

TIME, RES =  0.2052691947202211       2.4399115843842800E-06
-1.9503064291215289E-05  2.2635817943039704E-05
TIME, RES =  0.2057649732439289       2.2472561511666100E-06
-1.8791530976058901E-05  2.2289555254406679E-05
TIME, RES =  0.2062607517676368       1.8307841617715148E-06
-1.7494805797935165E-05  2.1708640520090583E-05
TIME, RES =  0.2067565302913446       1.5658142074603809E-06
-1.6595126988427911E-05  2.1286397740211127E-05

ITERATION   6

STATE CORRECTIONS
 0.10507203167E-04-0.35129799128E-05 0.26891823228E-04-0.21070100278E-03
-0.27545921954E-03-0.14811362808E-02 0.31840294636E-01-0.88195695331E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13269676645E+00-0.57728966290E+00 0.82459656653E+00 0.12519374688E-01
 0.53855486920E-01 0.27842131885E+00 0.14587627758E+00 0.69725391478E+01

TIME, RES =  0.2052691947202211       1.1875026073582262E-06
-7.5781260988061483E-06  1.1193513702822955E-05
TIME, RES =  0.2057649732439289       1.1219403288076357E-06
-7.3673571112653491E-06  1.1063438363573264E-05
TIME, RES =  0.2062607517676368       8.2446289273452322E-07
-6.5498630418270309E-06  1.0690255173556595E-05
TIME, RES =  0.2067565302913446       6.7040174583921086E-07
-6.1078650367418774E-06  1.0467294250443571E-05

ITERATION   7

STATE CORRECTIONS
 0.45282446528E-05-0.17805747867E-05 0.11500604328E-04-0.84986911286E-04
-0.10985697500E-03-0.70350535860E-03 0.29969113786E-01-0.86443911412E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13269223820E+00-0.57729144348E+00 0.82460806714E+00 0.12434387777E-01
 0.53745629945E-01 0.27771781349E+00 0.17584539137E+00 0.61081000337E+01

TIME, RES =  0.2052691947202211       6.2683829205045782E-07
-2.7210963581270065E-06  6.2412938401090745E-06
TIME, RES =  0.2057649732439289       6.3119578980197488E-07
-2.7397270758733328E-06  6.2148962306773914E-06
TIME, RES =  0.2062607517676368       3.9987773675642657E-07
-2.1415962865867222E-06  5.9414602582019693E-06
TIME, RES =  0.2067565302913446       3.0824053609546098E-07
-1.9089668261140780E-06  5.8143387694498117E-06
```

```
ITERATION   8

STATE CORRECTIONS
 0.27462066960E-05-0.16196496688E-05 0.64930225075E-05-0.49263870709E-04
-0.52349352002E-04-0.42349228180E-03 0.33989498929E-01-0.84834004228E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268949200E+00-0.57729306312E+00 0.82461456016E+00 0.12385123906E-01
 0.53693280593E-01 0.27729432121E+00 0.20983489030E+00 0.52597599914E+01

TIME, RES =  0.2052691947202211       3.2662703369767954E-07
-4.8302972865954530E-07  3.2730765318900179E-06
TIME, RES =  0.2057649732439289       3.7619291826107570E-07
-6.3535650890456097E-07  3.3109929015728845E-06
TIME, RES =  0.2062607517676368       1.8815782515391177E-07
-1.6575278327657372E-07  3.0998604212129077E-06
TIME, RES =  0.2067565302913446       1.3790726816065302E-07
-5.6545598259294394E-08  3.0330607325157466E-06

ITERATION   9

STATE CORRECTIONS
 0.17456386769E-05-0.13243255458E-05 0.38376762756E-05-0.30046467828E-04
-0.23521096371E-04-0.26020761701E-03 0.39101285821E-01-0.79200343418E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268774636E+00-0.57729438745E+00 0.82461839784E+00 0.12355077438E-01
 0.53669759497E-01 0.27703411359E+00 0.24893617612E+00 0.44677565572E+01

TIME, RES =  0.2052691947202211       1.6650548079200078E-07
 5.4025141882352301E-07  1.4134803224541415E-06
TIME, RES =  0.2057649732439289       2.4604540449502110E-07
 3.0673904155564813E-07  1.4929103238081609E-06
TIME, RES =  0.2062607517676368       8.7467600273782864E-08
 6.9658850321285826E-07  1.3227565739404312E-06
TIME, RES =  0.2067565302913446       6.6186762237574470E-08
 7.2739946194166905E-07  1.2964309033525190E-06

ITERATION  10

STATE CORRECTIONS
 0.10603764230E-05-0.10072700699E-05 0.21323986734E-05-0.17123684359E-04
-0.65942917516E-05-0.14797275235E-03 0.42631230537E-01-0.68538704227E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268668598E+00-0.57729539472E+00 0.82462053023E+00 0.12337953754E-01
 0.53663165205E-01 0.27688614084E+00 0.29156740666E+00 0.37823695150E+01

TIME, RES =  0.2052691947202211       8.9963929583714020E-08
 8.8821583910103641E-07  3.0212114698890957E-07
TIME, RES =  0.2057649732439289       1.8883654445689402E-07
 6.0753817388015108E-07  4.0744013396487588E-07
```

```
TIME, RES =  0.2062607517676368       5.0262238093790046E-08
 9.4850237492005718E-07  2.6388474264060768E-07
TIME, RES =  0.2067565302913446       4.9681373959220565E-08
 9.2863658163189200E-07  2.6489442100929850E-07

 ITERATION  11

 STATE CORRECTIONS
  0.58996260430E-06-0.70578848871E-06 0.10482445708E-05-0.85237178722E-05
  0.23280853620E-05-0.73132681392E-04 0.41273070469E-01-0.53394777626E+00

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268609602E+00-0.57729610051E+00 0.82462157848E+00 0.12329430036E-01
  0.53665493290E-01 0.27681300816E+00 0.33284047713E+00 0.32484217387E+01

TIME, RES =  0.2052691947202211       6.0640900689801214E-08
 8.8837447198564234E-07 -3.0461209196697148E-07
TIME, RES =  0.2057649732439289       1.7146500902454065E-07
 5.8217127352211406E-07 -1.8406224897016266E-07
TIME, RES =  0.2062607517676368       4.6319588387078170E-08
 8.9374617534554446E-07 -3.1083238671847369E-07
TIME, RES =  0.2067565302913446       6.0664519019315577E-08
 8.4056694926726294E-07 -2.9146114530198552E-07

 ITERATION  12

 STATE CORRECTIONS
  0.28998682333E-06-0.43840345484E-06 0.43033729400E-06-0.34599302786E-05
  0.51201809984E-05-0.29268319344E-04 0.32436272978E-01-0.36206691855E+00

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268580603E+00-0.57729653891E+00 0.82462200882E+00 0.12325970106E-01
  0.53670613471E-01 0.27678373984E+00 0.36527675010E+00 0.28863548202E+01

TIME, RES =  0.2052691947202211       5.3933016830320923E-08
 7.6769076678973036E-07 -5.9613180070083871E-07
TIME, RES =  0.2057649732439289       1.7188626821873498E-07
 4.4815825739341619E-07 -4.6708128476069533E-07
TIME, RES =  0.2062607517676368       5.5687559397110675E-08
 7.4162944524447383E-07 -5.8344026510059877E-07
TIME, RES =  0.2067565302913446       8.0809288816041658E-08
 6.6552803729136301E-07 -5.5173250532170037E-07

 ITERATION  13

 STATE CORRECTIONS
  0.12376820450E-06-0.23383871789E-06 0.14182331423E-06-0.10582554321E-05
  0.42769895741E-05-0.87921392544E-05 0.18714176057E-01-0.20847732406E+00

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268568226E+00-0.57729677275E+00 0.82462215064E+00 0.12324911851E-01
  0.53674890461E-01 0.27677494770E+00 0.38399092616E+00 0.26778774961E+01
```

```
TIME, RES =  0.2052691947202211       5.4631764945245465E-08
 6.5492717438164760E-07 -7.1744792520811451E-07
TIME, RES =  0.2057649732439289       1.7687019582268704E-07
 3.2776504704790099E-07 -5.8363829763541375E-07
TIME, RES =  0.2062607517676368       6.6707010193400862E-08
 6.0900014425824267E-07 -6.9339883737584707E-07
TIME, RES =  0.2067565302913446       9.9621225546631109E-08
 5.1602805162254128E-07 -6.5324469727956469E-07

 ITERATION  14

 STATE CORRECTIONS
  0.48279953112E-07-0.10622107547E-06 0.42912852018E-07-0.24629428053E-06
  0.23639967496E-05-0.20809080210E-05 0.74687175556E-02-0.10206843599E+00

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268563398E+00-0.57729687897E+00 0.82462219355E+00 0.12324665556E-01
  0.53677254458E-01 0.27677286679E+00 0.39145964372E+00 0.25758090601E+01

TIME, RES =  0.2052691947202211       5.5829807332763437E-08
 5.9146098457807383E-07 -7.6395602596757151E-07
TIME, RES =  0.2057649732439289       1.8063973156179358E-07
 2.5940551456882233E-07 -6.2741481302031055E-07
TIME, RES =  0.2062607517676368       7.4385618331618275E-08
 5.3222403051389477E-07 -7.3303826689241625E-07
TIME, RES =  0.2067565302913446       1.1254937370974716E-07
 4.2729634863736266E-07 -6.8733724140068730E-07

 ITERATION  15

 STATE CORRECTIONS
  0.19346769542E-07-0.40214963482E-07 0.19492436541E-07-0.83724255878E-07
  0.87726966731E-06-0.73216464608E-06 0.21295397734E-02-0.43621235338E-01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268561464E+00-0.57729691919E+00 0.82462221305E+00 0.12324581832E-01
  0.53678131727E-01 0.27677213463E+00 0.39358918349E+00 0.25321878248E+01

TIME, RES =  0.2052691947202211       5.6175996687723995E-08
 5.6892659694440795E-07 -7.8270282868886376E-07
TIME, RES =  0.2057649732439289       1.8236455312514721E-07
 2.3393937437221624E-07 -6.4470544292039023E-07
TIME, RES =  0.2062607517676368       7.8271107317728905E-08
 5.0176536431356666E-07 -7.4805096064722854E-07
TIME, RES =  0.2067565302913446       1.1937865856959107E-07
 3.8977673533224788E-07 -6.9924834877888031E-07

 ITERATION  16

 STATE CORRECTIONS
  0.81211831085E-08-0.12598958297E-07 0.11723010580E-07-0.54618731079E-07
  0.19929757142E-06-0.44818448634E-06 0.48828749549E-03-0.16912789047E-01
```

```
CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268560652E+00-0.57729693179E+00 0.82462222477E+00 0.12324527213E-01
 0.53678331025E-01 0.27677168644E+00 0.39407747098E+00 0.25152750357E+01


LAST PASS RESIDUALS:

TIME, RES =  0.2052691947202211       5.6140200099807913E-08
 5.6481546528708293E-07 -7.9072711509087235E-07
TIME, RES =  0.2057649732439289       1.8295497461462418E-07
 2.2835104179774746E-07 -6.5205806173973535E-07
TIME, RES =  0.2062607517676368       7.9850859813213049E-08
 4.9374274563307452E-07 -7.5435033458637335E-07
TIME, RES =  0.2067565302913446       1.2231124063566412E-07
 3.7835951577358529E-07 -7.0411215258303628E-07

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.1079787E-07-0.1225581E-07 0.1772386E-07-0.1447159E-06 0.1602750E-06
-0.2496890E-06-0.2348063E-08-0.6429218E-06

-0.1225581E-07 0.2124281E-07-0.1493236E-07 0.1622052E-06-0.2801692E-06
 0.2050094E-06 0.1983228E-06-0.3293695E-06

 0.1772386E-07-0.1493236E-07 0.3278480E-07-0.2391059E-06 0.1931286E-06
-0.4677527E-06 0.3945482E-06-0.2743141E-05

-0.1447159E-06 0.1622052E-06-0.2391059E-06 0.3139827E-05-0.3397113E-05
 0.5404993E-05 0.3222079E-05 0.1483250E-05

 0.1602750E-06-0.2801692E-06 0.1931286E-06-0.3397113E-05 0.5910122E-05
-0.4142106E-05-0.2041358E-04 0.6798370E-04

-0.2496890E-06 0.2050094E-06-0.4677527E-06 0.5404993E-05-0.4142106E-05
 0.1084129E-04-0.5382845E-04 0.2265523E-03

-0.2348063E-08 0.1983228E-06 0.3945482E-06 0.3222079E-05-0.2041358E-04
-0.5382845E-04 0.1185246E-01-0.4420482E-01

-0.6429218E-06-0.3293695E-06-0.2743141E-05 0.1483250E-05 0.6798370E-04
 0.2265523E-03-0.4420482E-01 0.1650555E+00


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  0.3044248994617925     -4.9224095739486273E-06
 1.4540944553331769E-05 -4.9643154554079860E-06
```

```
TIME, RES =  0.3049206779855004      -4.9721691519066802E-06
 1.4696157399407550E-05 -5.0135599591327740E-06
TIME, RES =  0.3054164565092082      -5.0221998129096335E-06
 1.4852411764687545E-05 -5.0631341166340604E-06
TIME, RES =  0.3059122350329161      -5.0725020753206174E-06
 1.5009711319291519E-05 -5.1130388901976520E-06

 SIGMA IN-TRACK =  0.1435904620022E-03


 ITERATION   1

 STATE CORRECTIONS
 -0.57324734695E-06-0.56321398532E-05-0.17989634111E-04-0.21417682370E-04
 -0.13433555640E-03-0.39669292481E-03-0.58351998798E-03-0.33130827796E-02

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13052106186E+00-0.56787575007E+00 0.85331332950E+00 0.32561688754E-01
  0.14171091146E+00 0.30575358323E+00 0.39349395100E+00 0.25119619529E+01


 LAST PASS RESIDUALS:

TIME, RES =  0.3044248994617925      -8.0159456850026345E-10
 8.6775643226566501E-08 -1.0527850138886130E-07
TIME, RES =  0.3049206779855004      -7.9473033709476226E-10
 8.3276906148732621E-08 -1.0112640433379561E-07
TIME, RES =  0.3054164565092082      -8.3369972037061757E-10
 8.0035574112624630E-08 -9.7036861212984604E-08
TIME, RES =  0.3059122350329161      -9.1863733286956517E-10
 7.7053279368133332E-08 -9.3010209784027964E-08

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.1585192E-07-0.1681237E-07 0.2596656E-07 0.1645801E-06-0.1820072E-06
 0.2606930E-06 0.2744042E-05-0.9171288E-05

-0.1681237E-07 0.2912212E-07-0.2004902E-07-0.1710534E-06 0.3030319E-06
-0.1780557E-06 0.9632062E-05-0.3328872E-04

 0.2596656E-07-0.2004902E-07 0.4759651E-07 0.2716153E-06-0.2266942E-06
 0.4794270E-06 0.8102480E-05-0.2746308E-04

 0.1645801E-06-0.1710534E-06 0.2716153E-06 0.5013273E-05-0.5737872E-05
 0.7895331E-05 0.2669922E-04-0.7943438E-04

-0.1820072E-06 0.3030319E-06-0.2266942E-06-0.5737872E-05 0.9946917E-05
-0.6554189E-05 0.4981182E-04-0.1722300E-03
```

```
 0.2606930E-06-0.1780557E-06 0.4794270E-06 0.7895331E-05-0.6554189E-05
 0.1627753E-04 0.1104036E-02-0.3763694E-02

 0.2744042E-05 0.9632062E-05 0.8102480E-05 0.2669922E-04 0.4981182E-04
 0.1104036E-02 0.5443716E+00-0.1873802E+01

-0.9171288E-05-0.3328872E-04-0.2746308E-04-0.7943438E-04-0.1722300E-03
-0.3763694E-02-0.1873802E+01 0.6450348E+01


REGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  0.3292138256471854      -5.9319263612245265E-08
 2.6143103615261509E-07  1.8594920769832157E-08
TIME, RES =  0.3297096041708932      -6.1775796811325279E-08
 2.7299977201078462E-07  1.9164278108885213E-08
TIME, RES =  0.3302053826946011      -6.4285348488901661E-08
 2.8492028741888831E-07  1.9651562793709942E-08
TIME, RES =  0.3307011612183089      -6.6848074020686710E-08
 2.9719476857259508E-07  2.0056325045203494E-08

 SIGMA IN-TRACK =  0.1394791557341E-03


 ITERATION   1

 STATE CORRECTIONS
  0.16323901097E-06-0.26782727494E-06 0.56066876744E-07-0.14749691843E-05
 -0.17008749189E-04-0.27866994489E-04 0.14997277274E-02-0.54661823943E-02

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12963595979E+00-0.56402474404E+00 0.86102713523E+00 0.38941263446E-01
  0.16946091568E+00 0.31686960999E+00 0.39499367872E+00 0.25064957705E+01


 LAST PASS RESIDUALS:

TIME, RES =  0.3292138256471854      -6.5864208531607460E-10
 5.9084605097403653E-08 -8.7929414083198765E-08
TIME, RES =  0.3297096041708932      -6.4851879422178627E-10
 5.6725218244846332E-08 -8.4526881632962514E-08
TIME, RES =  0.3302053826946011      -7.1473549301259709E-10
 5.4911018065872952E-08 -8.1252781530816165E-08
TIME, RES =  0.3307011612183089      -8.5712947850424825E-10
 5.3642841335577174E-08 -7.8107164319662914E-08

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA
```

```
 COVARIANCE MATRIX AT EPOCH IS:
 0.1678104E-07-0.1731438E-07 0.2760746E-07 0.1482814E-06-0.1705411E-06
 0.2307402E-06 0.3595634E-05-0.1166867E-04

-0.1731438E-07 0.3062083E-07-0.2019684E-07-0.1553359E-06 0.2600475E-06
-0.1577965E-06 0.1396917E-04-0.4591884E-04

 0.2760746E-07-0.2019684E-07 0.5088432E-07 0.2427718E-06-0.2246380E-06
 0.4242869E-06 0.1439888E-04-0.4709813E-04

 0.1482814E-06-0.1553359E-06 0.2427718E-06 0.3596756E-05-0.4084287E-05
 0.5493277E-05-0.5341215E-04 0.1825690E-03

-0.1705411E-06 0.2600475E-06-0.2246380E-06-0.4084287E-05 0.7460535E-05
-0.4999785E-05-0.2825622E-03 0.9302488E-03

 0.2307402E-06-0.1577965E-06 0.4242869E-06 0.5493277E-05-0.4999785E-05
 0.1047509E-04 0.3844328E-03-0.1236842E-02

 0.3595634E-05 0.1396917E-04 0.1439888E-04-0.5341215E-04-0.2825622E-03
 0.3844328E-03 0.3208594E+00-0.1053370E+01

-0.1166867E-04-0.4591884E-04-0.4709813E-04 0.1825690E-03 0.9302488E-03
-0.1236842E-02-0.1053370E+01 0.3458475E+01
```

Land Base Sensor

NONLINEAR BAYES FILTER

```
INITIAL STATE VECTOR :
-0.13260000000E+00-0.57700000000E+00 0.80590000000E+00-0.10230000000E-02
-0.44490000000E-02 0.62640000000E-02 0.37380000000E+00 0.34660000000E+01
INITIAL TIME : 0.002000 # OF DATA POINTS :     0
MAX LS ITERATIONS :        20  # OF BAYES CHUNKS :    50
MAX BAYES ITERATIONS  :    20  RANK OF P  :          8
BETA MATRIX =  1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000

FIRST PASS RESIDUALS:

TIME, RES =  2.4957785237078558E-03  1.8448517862253849E-05
-2.3190001380613001E-04  3.1130970579643438E-05
TIME, RES =  2.9915570474157117E-03  1.8448704051893944E-05
-2.3192109328873389E-04  3.1158639407858138E-05
TIME, RES =  3.4873355711235675E-03  1.8448870259833716E-05
-2.3194419709715586E-04  3.1186356499967455E-05
TIME, RES =  3.9831140948314234E-03  1.8449014879344527E-05
-2.3196945539660785E-04  3.1214121150570373E-05

ITERATION   1

STATE CORRECTIONS
-0.11314297351E-04 0.30462368608E-04 0.28280026399E-04 0.40358834297E-06
-0.13249647690E-06 0.12799220656E-04 0.10156564167E-03-0.78197015682E-03

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13261131430E+00-0.57696953763E+00 0.80592828003E+00-0.10225964117E-02
-0.44491324965E-02 0.62767992207E-02 0.37390156564E+00 0.34652180298E+01

TIME, RES =  2.4957785237078558E-03 -4.5373859981767684E-09
-2.5305657902485734E-08  6.1648782606615882E-10
TIME, RES =  2.9915570474157117E-03 -4.5379407419277040E-09
-2.5364198408261984E-08  6.9115368810102051E-10
TIME, RES =  3.4873355711235675E-03 -4.5387530643592466E-09
-2.5427572269975940E-08  7.5668833005027025E-10
TIME, RES =  3.9831140948314234E-03 -4.5398215013647825E-09
-2.5496095013011200E-08  8.1380740341030489E-10

ITERATION   2

STATE CORRECTIONS
 0.52198591303E-08 0.24406562612E-08 0.22459833969E-08 0.20028238320E-08
-0.40427085434E-08 0.34472377184E-07-0.84575884981E-04 0.75924844793E-03

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13261130908E+00-0.57696953519E+00 0.80592828227E+00-0.10225944088E-02
-0.44491365392E-02 0.62768336930E-02 0.37381698976E+00 0.34659772783E+01
```

```
 LAST PASS RESIDUALS:

TIME, RES =  2.4957785237078558E-03  2.0303203562832550E-14
 4.4408920985006262E-16  1.5671214326218319E-12
TIME, RES =  2.9915570474157117E-03  8.3044682241961709E-14
-2.2648549702353193E-14  6.0433134191351101E-12
TIME, RES =  3.4873355711235675E-03  1.8791218581171165E-13
-6.1950444774083735E-14  1.3519666289263377E-11
TIME, RES =  3.9831140948314234E-03  3.3503408380930466E-13
-1.1912693054227930E-13  2.4005847656938251E-11

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.9952970E-10 0.4346840E-09-0.2433596E-09-0.1774353E-07-0.7721903E-07
 0.1056279E-06-0.1163509E-03 0.1017210E-02

 0.4346840E-09 0.2151919E-08-0.1429787E-08-0.7749480E-07-0.3398025E-06
 0.4650144E-06-0.5240531E-03 0.4589980E-02

-0.2433596E-09-0.1429787E-08 0.3188217E-08 0.1056544E-06 0.4634336E-06
-0.6679942E-06 0.7366104E-03-0.6451023E-02

-0.1774353E-07-0.7749480E-07 0.1056544E-06 0.6492580E-05 0.2825354E-04
-0.3991971E-04 0.5108447E-01-0.4478712E+00

-0.7721903E-07-0.3398025E-06 0.4634336E-06 0.2825354E-04 0.1229982E-03
-0.1737858E-03 0.2225964E+00-0.1951713E+01

 0.1056279E-06 0.4650144E-06-0.6679942E-06-0.3991971E-04-0.1737858E-03
 0.2461941E-03-0.3148609E+00 0.2760645E+01

-0.1163509E-03-0.5240531E-03 0.7366104E-03 0.5108447E-01 0.2225964E+00
-0.3148609E+00 0.4758258E+03-0.4185478E+04

 0.1017210E-02 0.4589980E-02-0.6451023E-02-0.4478712E+00-0.1951713E+01
 0.2760645E+01-0.4185478E+04 0.3681900E+05


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  2.7284704709100647E-02  5.8865527036155996E-11
-2.1008750294981837E-12  4.1108263182376703E-09
TIME, RES =  2.7780483232808503E-02  6.1308354226685680E-11
-3.4650060598551136E-13  4.2788280505551279E-09
TIME, RES =  2.8276261756516359E-02  6.3805991740162327E-11
 1.6535661728767082E-12  4.4504220621244261E-09
TIME, RES =  2.8772040280224215E-02  6.6358824685197604E-11
```

```
3.8953285041998242E-12  4.6256138329370255E-09

SIGMA IN-TRACK =  0.1666674317104E-04


ITERATION   1

STATE CORRECTIONS
-0.87678398610E-10-0.38148129523E-09 0.53389609320E-09-0.73890898778E-08
-0.32152330156E-07 0.44096371176E-07 0.17303193378E-07 0.52380045297E-05

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13264929664E+00-0.57713481269E+00 0.80616992861E+00-0.20694915254E-02
-0.90040100787E-02 0.13751501021E-01 0.37381700706E+00 0.34659825163E+01


LAST PASS RESIDUALS:

TIME, RES =  2.7284704709100647E-02 -1.4918621893400541E-16
-1.1102230246251565E-16  5.8878249498128810E-14
TIME, RES =  2.7780483232808503E-02  3.6429192995512949E-16
-6.3282712403633923E-15  8.7695475881055529E-14
TIME, RES =  2.8276261756516359E-02  1.2594092435591620E-15
-1.1102230246251565E-15  1.4577922202718696E-13
TIME, RES =  2.8772040280224215E-02  2.5708601913976281E-15
-4.9960036108132044E-15  2.2435352187155644E-13

CONVERGENCE ACHIEVED.
IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
DECLARIUM EST ESTIMATIA


COVARIANCE MATRIX AT EPOCH IS:
0.1021808E-09 0.4459627E-09-0.1873908E-09-0.1601955E-07-0.6973250E-07
0.9609681E-07-0.8787998E-04 0.6990143E-03

0.4459627E-09 0.2278202E-08-0.1294304E-08-0.6978741E-07-0.3118651E-06
0.4301474E-06-0.3882012E-03 0.3093026E-02

-0.1873908E-09-0.1294304E-08 0.3486790E-08 0.9547385E-07 0.4270635E-06
-0.6994367E-06 0.5916256E-03-0.4712534E-02

-0.1601955E-07-0.6978741E-07 0.9547385E-07 0.5549536E-05 0.2414816E-04
-0.3696024E-04 0.3826282E-01-0.3052991E+00

-0.6973250E-07-0.3118651E-06 0.4270635E-06 0.2414816E-04 0.1053920E-03
-0.1612732E-03 0.1667023E+00-0.1330314E+01

0.9609681E-07 0.4301474E-06-0.6994367E-06-0.3696024E-04-0.1612732E-03
0.2513625E-03-0.2576357E+00 0.2055844E+01

-0.8787998E-04-0.3882012E-03 0.5916256E-03 0.3826282E-01 0.1667023E+00
-0.2576357E+00 0.3126587E+03-0.2503272E+04
```

```
 0.6990143E-03 0.3093026E-02-0.4712534E-02-0.3052991E+00-0.1330314E+01
 0.2055844E+01-0.2503272E+04 0.2004372E+05


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  5.2073630894493458E-02  4.9264065049570149E-13
-1.7560397580496101E-12  3.3055645026758285E-11
TIME, RES =  5.2569409418201314E-02  5.1326998207201768E-13
-1.8227641618295820E-12  3.4413484215067847E-11
TIME, RES =  5.3065187941909170E-02  5.3445095571369450E-13
-1.8967050152696174E-12  3.5808130766090684E-11
TIME, RES =  5.3560966465617026E-02  5.5613499916340459E-13
-1.9659829320062272E-12  3.7227602944778226E-11

 SIGMA IN-TRACK =  0.1668112660909E-04


 ITERATION   1

 STATE CORRECTIONS
 -0.65500222064E-12-0.28503051805E-11 0.44572870523E-11-0.54810502834E-10
 -0.24036247152E-09 0.37429090367E-09-0.20000152684E-09 0.37782548127E-07

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13271389418E+00-0.57741586575E+00 0.80664307130E+00-0.31096134892E-02
 -0.13529406067E-01 0.25228387479E-01 0.37381700686E+00 0.34659825541E+01


 LAST PASS RESIDUALS:

TIME, RES =  5.2073630894493458E-02 -1.7000290064572710E-16
-3.3306690738754696E-16  3.0765320846448674E-14
TIME, RES =  5.2569409418201314E-02 -2.0469737016526324E-16
 2.1094237467877974E-15  2.4917568008930857E-14
TIME, RES =  5.3065187941909170E-02 -1.5959455978986625E-16
-1.5543122344752192E-15  2.6579433098916638E-14
TIME, RES =  5.3560966465617026E-02 -9.7144514654701197E-17
-1.1102230246251565E-16  2.3927040904148100E-14

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.9582257E-10 0.4170304E-09-0.1194127E-09-0.1225665E-07-0.5332519E-07
 0.7963670E-07-0.5843050E-04 0.4184784E-03

 0.4170304E-09 0.2216048E-08-0.1091350E-08-0.5336797E-07-0.2459154E-06
```

```
 0.3664448E-06-0.2559888E-03 0.1836552E-02

-0.1194127E-09-0.1091350E-08 0.3787536E-08 0.7888053E-07 0.3627788E-06
-0.7576789E-06 0.4787021E-03-0.3432403E-02

-0.1225665E-07-0.5336797E-07 0.7888053E-07 0.3887359E-05 0.1691402E-04
-0.3133359E-04 0.2519131E-01-0.1810560E+00

-0.5332519E-07-0.2459154E-06 0.3627788E-06 0.1691402E-04 0.7429511E-04
-0.1373414E-03 0.1096935E+00-0.7885532E+00

 0.7963670E-07 0.3664448E-06-0.7576789E-06-0.3133359E-04-0.1373414E-03
 0.2668306E-03-0.2095178E+00 0.1505928E+01

-0.5843050E-04-0.2559888E-03 0.4787021E-03 0.2519131E-01 0.1096935E+00
-0.2095178E+00 0.1959883E+03-0.1413871E+04

 0.4184784E-03 0.1836552E-02-0.3432403E-02-0.1810560E+00-0.7885532E+00
 0.1505928E+01-0.1413871E+04 0.1020066E+05


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  7.6862557079886247E-02  4.4061976289810900E-15
-5.3845816694320092E-14  3.1765041985654108E-13
TIME, RES =  7.7358335603594102E-02  4.6074255521943996E-15
-5.1847415249994810E-14  3.3374171481970194E-13
TIME, RES =  7.7854114127301957E-02  4.7739590058881731E-15
-6.0840221749458578E-14  3.4938718584953676E-13
TIME, RES =  7.8349892651009813E-02  4.8919202022545960E-15
-6.3393734706096438E-14  3.6626604527079110E-13

 SIGMA IN-TRACK =   0.1669833891639E-04


 ITERATION   1

 STATE CORRECTIONS
 -0.46465859151E-14-0.20378244274E-13 0.42944297481E-13-0.36379065245E-12
 -0.28772496149E-11 0.52258055059E-11 0.28226532239E-10 0.13761676262E-09

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13280006712E+00-0.57779078973E+00 0.80746713135E+00-0.37386480452E-02
 -0.16266229783E-01 0.42310351301E-01 0.37381700689E+00 0.34659825542E+01


 LAST PASS RESIDUALS:

TIME, RES =  7.6862557079886247E-02 -1.4571677198205180E-16
-2.2204460492503131E-16  2.0785456689154103E-14
TIME, RES =  7.7358335603594102E-02 -9.3675067702747583E-17
```

```
 4.1078251911130792E-15  1.8901546994243290E-14
TIME, RES =  7.7854114127301957E-02 -7.2858385991025898E-17
-2.5535129566378600E-15  1.6233542288190961E-14
TIME, RES =  7.8349892651009813E-02 -1.0755285551056204E-16
-2.6645352591003757E-15  1.4547391069541504E-14

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.8717017E-10 0.3751040E-09-0.3326689E-10-0.8105970E-08-0.3507808E-07
 0.5617388E-07-0.3302409E-04 0.2103316E-03

 0.3751040E-09 0.2088885E-08-0.8117545E-09-0.3514260E-07-0.1713959E-06
 0.2724029E-06-0.1439905E-03 0.9189760E-03

-0.3326689E-10-0.8117545E-09 0.4034138E-08 0.5557304E-07 0.2689732E-06
-0.8119042E-06 0.3781973E-03-0.2411306E-02

-0.8105970E-08-0.3514260E-07 0.5557304E-07 0.2134409E-05 0.9275636E-05
-0.2299924E-04 0.1405391E-01-0.8987830E-01

-0.3507808E-07-0.1713959E-06 0.2689732E-06 0.9275636E-05 0.4144713E-04
-0.1017038E-03 0.6116449E-01-0.3912732E+00

 0.5617388E-07 0.2724029E-06-0.8119042E-06-0.2299924E-04-0.1017038E-03
 0.2816984E-03-0.1652431E+00 0.1056772E+01

-0.3302409E-04-0.1439905E-03 0.3781973E-03 0.1405391E-01 0.6116449E-01
-0.1652431E+00 0.1160075E+03-0.7449684E+03

 0.2103316E-03 0.9189760E-03-0.2411306E-02-0.8987830E-01-0.3912732E+00
 0.1056772E+01-0.7449684E+03 0.4784510E+04


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  0.1016514832652790     -2.7755575615628914E-16
-5.6621374255882984E-15  1.6615181452905858E-14
TIME, RES =  0.1021472617889869     -1.5612511283791264E-16
-4.9960036108132044E-15  2.3573157315048832E-14
TIME, RES =  0.1026430403126947     -3.4000580129145419E-16
-6.8833827526759706E-15  1.9579823873350222E-14
TIME, RES =  0.1031388188364026     -3.1225022567582528E-16
-7.9936057773011271E-15  2.3337234922315986E-14

 SIGMA IN-TRACK =  0.1672010053966E-04
```

```
ITERATION   1

STATE CORRECTIONS
 0.18790925274E-15 0.69812975344E-15-0.30507751996E-15 0.63520937713E-13
-0.71274434228E-12 0.13069354146E-11-0.30967718809E-11 0.68009983775E-10

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13289245339E+00-0.57819274689E+00 0.80879758045E+00-0.35486447744E-02
-0.15439557466E-01 0.66263123332E-01 0.37381700689E+00 0.34659825543E+01


LAST PASS RESIDUALS:

TIME, RES =  0.1016514832652790     -1.4918621893400541E-16
-2.6645352591003757E-15  1.6184970030863610E-14
TIME, RES =  0.1021472617889869      6.9388939039072284E-18
-1.2212453270876722E-15  1.8735013540549517E-14
TIME, RES =  0.1026430403126947     -1.4224732503009818E-16
-2.4424906541753444E-15  1.0231399061311208E-14
TIME, RES =  0.1031388188364026     -7.9797279894933126E-17
-2.8865798640254070E-15  9.5323055004925550E-15

CONVERGENCE ACHIEVED.
IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
DECLARIUM EST ESTIMATIA


COVARIANCE MATRIX AT EPOCH IS:
 0.8240535E-10 0.3440370E-09 0.6705432E-10-0.5233755E-08-0.2206845E-07
 0.2867981E-07-0.1477205E-04 0.8234582E-04

 0.3440370E-09 0.2001734E-08-0.4910335E-09-0.2214222E-07-0.1188648E-06
 0.1617684E-06-0.6408704E-04 0.3583440E-03

 0.6705432E-10-0.4910335E-09 0.4147672E-08 0.2841356E-07 0.1589995E-06
-0.8267293E-06 0.2807439E-03-0.1566964E-02

-0.5233755E-08-0.2214222E-07 0.2841356E-07 0.8951293E-06 0.3846952E-05
-0.1319633E-04 0.6148244E-02-0.3444676E-01

-0.2206845E-07-0.1188648E-06 0.1589995E-06 0.3846952E-05 0.1817094E-04
-0.5976963E-04 0.2673836E-01-0.1498773E+00

 0.2867981E-07 0.1617684E-06-0.8267293E-06-0.1319633E-04-0.5976963E-04
 0.2818795E-03-0.1218494E+00 0.6826465E+00

-0.1477205E-04-0.6408704E-04 0.2807439E-03 0.6148244E-02 0.2673836E-01
-0.1218494E+00 0.6328257E+02-0.3562099E+03

 0.8234582E-04 0.3583440E-03-0.1566964E-02-0.3444676E-01-0.1498773E+00
 0.6826465E+00-0.3562099E+03 0.2005346E+04
```

SIGMA IN-TRACK = 0.1683198864345E-04

LAST GOOD VALUES FOR THE MAIN STATE VECTOR

```
-0.13273855498E+00  -0.57752316076E+00   0.82341630055E+00   0.10999752541E-01
 0.47858075956E-01   0.26249364632E+00   0.37381700695E+00   0.34659825541E+01
```

NONLINEAR LS STAGING ESTIMATOR

INITIAL GUESS IS :

VE*M = 0.98000000000E+00 TSTAGE = 0.20450000000E+00

Q INVERSE, A 1 X 1 MATRIX, IS :

0.35296296671E+10

FIRST PASS RESIDUALS :

RESIDUAL = -0.60667594581E-06

RESIDUAL = -0.92077802844E-06

RESIDUAL = -0.13031513361E-05

RESIDUAL = -0.17642025803E-05

RESIDUAL = -0.23090783295E-05

P INVERSE MATRIX IS :

```
 0.14002829422E+04   0.49312284707E+06
 0.49312284707E+06   0.18685772084E+09
```

P MATRIX IS :

```
 0.10109413740E-01  -0.26679030780E-04
-0.26679030780E-04   0.75758387460E-07
```

ITERATION 1

STATE CORRECTIONS

```
-0.49716864143E+00   0.57629874606E-03

CURRENT STAGING STATE VECTOR
VE*M =  0.48283135857E+00  TSTAGE =  0.20507629875E+00


RESIDUAL =  0.17853352765E-06

RESIDUAL = -0.10514035089E-05

RESIDUAL = -0.13884175885E-05

RESIDUAL = -0.24531349446E-05

RESIDUAL = -0.28728138233E-05

P INVERSE MATRIX IS :

 0.12180637603E+04   0.52242272939E+06
 0.52242272939E+06   0.24277246028E+09

P MATRIX IS :

 0.10654070136E-01  -0.22926523022E-04
-0.22926523022E-04   0.53454731717E-07

ITERATION   2

STATE CORRECTIONS
 0.47353621114E-02  -0.89575794963E-04

CURRENT STAGING STATE VECTOR
VE*M =  0.48756672068E+00  TSTAGE =  0.20498672295E+00


RESIDUAL =  0.88336356369E-07

RESIDUAL = -0.96890627918E-06

RESIDUAL = -0.11355161182E-05

RESIDUAL = -0.20321193943E-05

RESIDUAL = -0.22859743536E-05

P INVERSE MATRIX IS :

 0.12452194472E+04   0.53014744828E+06
 0.53014744828E+06   0.24457180922E+09

P MATRIX IS :

 0.10412071273E-01  -0.22569784450E-04
```

```
-0.22569784450E-04   0.53012298006E-07

ITERATION   3

STATE CORRECTIONS
 0.37266855314E-03  -0.80978025335E-06

CURRENT STAGING STATE VECTOR
VE*M =  0.48793938923E+00  TSTAGE =  0.20498591317E+00


LAST PASS RESIDUALS :


RESIDUAL =  0.87657575102E-07

RESIDUAL = -0.96803343567E-06

RESIDUAL = -0.11331934210E-05

RESIDUAL = -0.20284486145E-05

RESIDUAL = -0.22810572622E-05

CONVERGENCE HAS BEEN ACHEIVED
COVARIENCE MATRIX :

 0.10412071273E-01  -0.22569784450E-04
-0.22569784450E-04   0.53012298006E-07

                        BEGIN ESTIMATION OF THE NEXT STAGE

                        STAGING OCCURED AT : 165.3854  SECONDS


 AT THE START OF THE NEXT STAGE THE

COVARIANCE MATRIX AT EPOCH IS:
0.7219619E-10 0.2292700E-09 0.2974801E-09-0.2272561E-08-0.4619524E-08
0.3999174E-07-0.3954023E-05 0.1285603E-04

0.2292700E-09 0.1341629E-08 0.4581231E-09-0.4564984E-08-0.4043245E-07
0.2292172E-06-0.1708788E-04 0.5560764E-04

0.2974801E-09 0.4581231E-09 0.4375583E-08 0.4098511E-07 0.2294855E-06
0.1374155E-05-0.1090551E-03 0.3546261E-03

-0.2272561E-08-0.4564984E-08 0.4098511E-07 0.1393559E-05 0.5387628E-05
0.2713599E-04-0.1887962E-02 0.6122962E-02

-0.4619524E-08-0.4043245E-07 0.2294855E-06 0.5387628E-05 0.2560498E-04
0.1117689E-03-0.8227548E-02 0.2667774E-01
```

```
 0.3999174E-07 0.2292172E-06 0.1374155E-05 0.2713599E-04 0.1117689E-03
 0.7318132E-03-0.5286334E-01 0.1712357E+00

-0.3954023E-05-0.1708788E-04-0.1090551E-03-0.1887962E-02-0.8227548E-02
-0.5286334E-01 0.7807597E+01-0.1266701E+02

 0.1285603E-04 0.5560764E-04 0.3546261E-03 0.6122962E-02 0.2667774E-01
 0.1712357E+00-0.1266701E+02 0.4767188E+02


 INITIAL STATE VECTOR IS:

 -0.13268644568E+00-0.57729644194E+00 0.82462090470E+00 0.12371639365E-01
  0.53826925127E-01 0.27759580461E+00 0.37381700695E+00 0.13052894335E+01

 FIRST PASS RESIDUALS:

TIME, RES =  0.2052691947202211      -1.0668006885511394E-07
 2.7666640292078881E-06 -2.5221229591175249E-06
TIME, RES =  0.2057649732439289      -6.6257463609414868E-08
 1.7092021478148922E-06 -1.5586033374444860E-06
TIME, RES =  0.2062607517676368      -1.4542719782920099E-07
 3.7284482460320234E-06 -3.4051571719666838E-06
TIME, RES =  0.2067565302913446      -1.1518067354748807E-07
 2.9312333728048756E-06 -2.6859194964224623E-06

 ITERATION   1

 STATE CORRECTIONS
  0.27947285933E-05 0.51659989201E-05 0.12463054502E-04-0.22407685694E-03
 -0.48035112741E-03-0.17931369721E-02-0.48885313317E+00 0.24325465391E+01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268365095E+00-0.57729127594E+00 0.82463336776E+00 0.12147562508E-01
  0.53346574000E-01 0.27580266764E+00-0.11503612622E+00 0.37378359726E+01

TIME, RES =  0.2052691947202211      -3.9398841313389998E-07
 6.8081716283918681E-05 -3.8993217985426229E-05
TIME, RES =  0.2057649732439289      -2.1515823409382073E-07
 6.1286736198762348E-05 -3.3391813777461561E-05
TIME, RES =  0.2062607517676368      -1.1626135429376738E-07
 5.6602695536311920E-05 -2.9724568292323209E-05
TIME, RES =  0.2067565302913446       1.3226517794201476E-07
 4.8134941661115427E-05 -2.2616540661754342E-05

 ITERATION   2

 STATE CORRECTIONS
  0.20213936525E-04 0.38004641141E-04 0.63198943909E-04-0.14703671177E-02
 -0.23317662547E-02-0.48829433057E-02 0.27525132797E+00-0.20838244703E+01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
```

```
-0.13266343702E+00-0.57725327130E+00 0.82469656670E+00 0.10677195390E-01
 0.51014807745E-01 0.27091972433E+00 0.16021520176E+00 0.16540115023E+01

TIME, RES =  0.2052691947202211       1.3095896438455878E-06
 4.7161928405947684E-04 -1.9525755752451571E-04
TIME, RES =  0.2057649732439289       1.3168111263200943E-06
 4.5129367802287934E-04 -1.8332666373848067E-04
TIME, RES =  0.2062607517676368       1.2191644338799268E-06
 4.3381358600924624E-04 -1.7399765232106637E-04
TIME, RES =  0.2067565302913446       1.2457809634505712E-06
 4.1328569900977996E-04 -1.6189509109424348E-04

 ITERATION   3

 STATE CORRECTIONS
 -0.17459945581E-04-0.33881940895E-04-0.48065121542E-04 0.13185494741E-02
  0.20905297535E-02 0.36778866682E-02-0.91053293263E-02 0.36026970955E+01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268089696E+00-0.57728715324E+00 0.82464850158E+00 0.11995744865E-01
  0.53105337499E-01 0.27459761100E+00 0.15110987243E+00 0.52567085978E+01

TIME, RES =  0.2052691947202211     -1.0996823459653859E-06
 1.4042464282282285E-04 -9.3573461807766188E-05
TIME, RES =  0.2057649732439289     -9.5380379165621587E-07
 1.3285719729894119E-04 -8.7349612804029980E-05
TIME, RES =  0.2062607517676368     -9.3802674196868940E-07
 1.2869421153727245E-04 -8.4233738722306164E-05
TIME, RES =  0.2067565302913446     -8.2358948817404243E-07
 1.2204451138764227E-04 -7.8851014942505357E-05

 ITERATION   4

 STATE CORRECTIONS
 -0.55228533587E-05-0.91595367698E-05-0.28322576164E-04 0.34490417081E-03
  0.61070190922E-03 0.24261349755E-02 0.99614878725E-01-0.11236709989E+01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.13268641981E+00-0.57729631277E+00 0.82462017900E+00 0.12340649035E-01
  0.53716039408E-01 0.27702374597E+00 0.25072475116E+00 0.41330375989E+01

TIME, RES =  0.2052691947202211       5.9345062804888604E-08
-3.9589450051558117E-08  2.1755918473844544E-06
TIME, RES =  0.2057649732439289       1.1223209894425934E-07
-1.4702245543229964E-06  3.4774821489025028E-06
TIME, RES =  0.2062607517676368       2.3137472028589290E-08
 7.5657393683314922E-07  1.4428978898560442E-06
TIME, RES =  0.2067565302913446       2.0685401072584320E-08
 7.4974219210766080E-07  1.4471472687358497E-06

 ITERATION   5

 STATE CORRECTIONS
```

0.12568873931E-06 0.34930011033E-06 0.11051634702E-05-0.10722929414E-04
-0.48360988195E-04-0.20456901780E-03 0.61990752066E-01-0.67310049094E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268629413E+00-0.57729596347E+00 0.82462128417E+00 0.12329926106E-01
0.53667678420E-01 0.27681917696E+00 0.31271550322E+00 0.34599371080E+01

TIME, RES = 0.2052691947202211 -5.9440643084690548E-08
5.0653607587269889E-06 -1.3858576813672403E-06
TIME, RES = 0.2057649732439289 6.9109882677720336E-09
3.2425434746130577E-06 2.3914259472798596E-07
TIME, RES = 0.2062607517676368 -7.0431791444197644E-08
5.1250496658772349E-06 -1.5158005831766852E-06
TIME, RES = 0.2067565302913446 -6.2858944677862683E-08
4.8215450664246262E-06 -1.2750228422965396E-06

ITERATION 6

STATE CORRECTIONS
0.71627086334E-07 0.98337463637E-07 0.48019908211E-06-0.62856487820E-05
-0.18654083950E-04-0.86701751338E-04 0.41876536390E-01-0.44912189366E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268622250E+00-0.57729586514E+00 0.82462176437E+00 0.12323640457E-01
0.53649024336E-01 0.27673247521E+00 0.35459203961E+00 0.30108152143E+01

TIME, RES = 0.2052691947202211 -8.5885428307425071E-08
7.0545995116688687E-06 -3.1119083535280064E-06
TIME, RES = 0.2057649732439289 -1.2761536424821252E-08
5.0152662687263927E-06 -1.2980526630066054E-06
TIME, RES = 0.2062607517676368 -8.2651255142035529E-08
6.6657929627123025E-06 -2.8501384153564215E-06
TIME, RES = 0.2067565302913446 -6.6925294705394434E-08
6.1145614104196611E-06 -2.3922799560646647E-06

ITERATION 7

STATE CORRECTIONS
-0.55958817895E-07-0.11205552011E-06-0.14511810623E-06 0.31615365692E-05
0.12608074536E-05-0.99131061084E-05 0.26784026054E-01-0.27761919539E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268627846E+00-0.57729597719E+00 0.82462161925E+00 0.12326801994E-01
0.53650285143E-01 0.27672256210E+00 0.38137606567E+00 0.27331960189E+01

TIME, RES = 0.2052691947202211 -9.2251456928343911E-08
5.9439568005359433E-06 -2.7864570481668472E-06
TIME, RES = 0.2057649732439289 -1.5145693154500206E-08
3.8531791556328798E-06 -9.1156972474961458E-07
TIME, RES = 0.2062607517676368 -7.9985648042402158E-08
5.4254391060881701E-06 -2.3782884488108809E-06
TIME, RES = 0.2067565302913446 -5.8120182999665815E-08
4.7689174730303208E-06 -1.8106078855120939E-06

```
ITERATION   8

STATE CORRECTIONS
-0.46740635328E-07-0.88975497940E-07-0.16009519675E-06 0.30069009510E-05
 0.33362773978E-05 0.47528880418E-05 0.12833686088E-01-0.14736778570E+00

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268632520E+00-0.57729606617E+00 0.82462145915E+00 0.12329808895E-01
 0.53653621421E-01 0.27672731499E+00 0.39420975175E+00 0.25858282332E+01

TIME, RES =  0.2052691947202211     -9.2568815009130834E-08
 4.9309574480238538E-06 -2.3423131804570577E-06
TIME, RES =  0.2057649732439289     -1.3169310766891762E-08
 2.8216741309039506E-06 -4.3980834889224482E-07
TIME, RES =  0.2062607517676368     -7.4756849342827447E-08
 4.3510427293336207E-06 -1.8567887637956559E-06
TIME, RES =  0.2067565302913446     -4.8662902121104912E-08
 3.6271241935503795E-06 -1.2171972008565273E-06

ITERATION   9

STATE CORRECTIONS
-0.21814542917E-07-0.41431189138E-07-0.77328410591E-07 0.14682438772E-05
 0.18168483112E-05 0.36187938125E-05 0.43035319706E-02-0.67198725485E-01

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268634701E+00-0.57729610760E+00 0.82462138182E+00 0.12331277139E-01
 0.53655438269E-01 0.27673093378E+00 0.39851328372E+00 0.25186295078E+01

TIME, RES =  0.2052691947202211     -9.2283821154226509E-08
 4.4501171104771586E-06 -2.1229694431953343E-06
TIME, RES =  0.2057649732439289     -1.1627204511699407E-08
 2.3274525933203449E-06 -2.0331170162257283E-07
TIME, RES =  0.2062607517676368     -7.1307596889724767E-08
 3.8268837719845905E-06 -1.5881245615801927E-06
TIME, RES =  0.2067565302913446     -4.2645423869308585E-08
 3.0564096480389935E-06 -9.0133315333411346E-07

ITERATION  10

STATE CORRECTIONS
-0.65324430519E-08-0.12885990430E-07-0.21565212354E-07 0.45648874025E-06
 0.53998262219E-06 0.97032282081E-06 0.10791526940E-02-0.27157126207E-01

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13268635355E+00-0.57729612049E+00 0.82462136026E+00 0.12331733627E-01
 0.53655978252E-01 0.27673190410E+00 0.39959243642E+00 0.24914723815E+01


LAST PASS RESIDUALS:

TIME, RES =  0.2052691947202211     -9.2311275984302554E-08
```

```
 4.3073794059322879E-06 -2.0644980302706416E-06
TIME, RES =  0.2057649732439289      -1.1049917609390336E-08
 2.1750744690907098E-06 -1.3449998229270524E-07
TIME, RES =  0.2062607517676368      -6.9787447175284623E-08
 3.6562676386520110E-06 -1.5011770738001737E-06
TIME, RES =  0.2067565302913446      -3.9838729780361515E-08
 2.8589302542014750E-06 -7.8844934725097026E-07

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.2156279E-10 0.5614992E-10 0.6614793E-10-0.3907751E-09-0.1048720E-08
-0.2106966E-08 0.8397057E-07-0.2854256E-06

 0.5614992E-10 0.3144568E-09 0.2744374E-10-0.9228647E-09-0.4783594E-08
-0.2693178E-08 0.1856897E-06-0.6369346E-06

 0.6614793E-10 0.2744374E-10 0.4326082E-09-0.1551008E-08-0.2667617E-08
-0.1553572E-07 0.8721276E-06-0.3032271E-05

-0.3907751E-09-0.9228647E-09-0.1551008E-08 0.1352736E-07 0.3195293E-07
 0.8530684E-07-0.5101951E-05 0.1784111E-04

-0.1048720E-08-0.4783594E-08-0.2667617E-08 0.3195293E-07 0.1372941E-06
 0.2036460E-06-0.1837260E-04 0.6495248E-04

-0.2106966E-08-0.2693178E-08-0.1553572E-07 0.8530684E-07 0.2036460E-06
 0.9744056E-06-0.8370782E-04 0.2972636E-03

 0.8397057E-07 0.1856897E-06 0.8721276E-06-0.5101951E-05-0.1837260E-04
-0.8370782E-04 0.1053589E-01-0.3802754E-01

-0.2854256E-06-0.6369346E-06-0.3032271E-05 0.1784111E-04 0.6495248E-04
 0.2972636E-03-0.3802754E-01 0.1373587E+00


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  0.3044248994617925      -1.0570388328901698E-05
 1.3911211906703080E-04 -1.0599243159586762E-04
TIME, RES =  0.3049206779855004      -1.0707369779264925E-05
 1.4052377534878868E-04 -1.0698789522938131E-04
TIME, RES =  0.3054164565092082      -1.0845476437117835E-05
 1.4194169065884221E-04 -1.0798661869996021E-04
TIME, RES =  0.3059122350329161      -1.0984712339595637E-05
 1.4336584008778352E-04 -1.0898857970778955E-04

 SIGMA IN-TRACK =  0.1694476316555E-04
```

```
ITERATION   1

STATE CORRECTIONS
-0.19247846794E-05-0.85956084254E-05-0.31247054716E-04-0.45637655153E-04
-0.20140437991E-03-0.65371760339E-03-0.49993159369E-02 0.88393356341E-02

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13052117389E+00-0.56787568869E+00 0.85331308619E+00 0.32569793558E-01
 0.14170672412E+00 0.30577277261E+00 0.39459312048E+00 0.25003117172E+01

TIME, RES =  0.3044248994617925      -1.3769985707234866E-09
 3.9087956427152903E-08  9.5910831056714163E-08
TIME, RES =  0.3049206779855004      -1.1810167219183931E-09
 3.0781548865377317E-08  9.5965416238574797E-08
TIME, RES =  0.3054164565092082      -7.3901854774627296E-10
 1.8631565068538691E-08  9.7980299972391660E-08
TIME, RES =  0.3059122350329161      -4.8075224357013724E-11
 2.6255264629071462E-09  1.0195451080954876E-07

ITERATION   2

STATE CORRECTIONS
-0.10722150156E-07-0.61757075247E-07-0.21764904283E-07 0.16577100941E-05
 0.70343076176E-05 0.73719177348E-05-0.64079980228E-02 0.22155013272E-01

CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
-0.13052118462E+00-0.56787575045E+00 0.85331306442E+00 0.32571451268E-01
 0.14171375842E+00 0.30578014452E+00 0.38818512246E+00 0.25224667305E+01


LAST PASS RESIDUALS:

TIME, RES =  0.3044248994617925       8.4121458791830417E-09
-2.8659860962232386E-07 -2.4110829931547895E-08
TIME, RES =  0.3049206779855004       8.1024207794488934E-09
-2.7763874843600433E-07 -2.2472149219590620E-08
TIME, RES =  0.3054164565092082       8.4841813643987685E-09
-2.7960959858575052E-07 -1.5227996991856729E-08
TIME, RES =  0.3059122350329161       9.5627215243587749E-09
-2.9249940558528920E-07 -2.4052717300521165E-09

CONVERGENCE ACHIEVED.
IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
DECLARIUM EST ESTIMATIA


COVARIANCE MATRIX AT EPOCH IS:
0.8404415E-10 0.1046792E-09 0.1566623E-09 0.9782895E-09 0.9452315E-09
0.2142694E-08 0.3587019E-06-0.1220101E-05

0.1046792E-09 0.4756074E-09-0.3150754E-10 0.1228656E-08 0.5207403E-08
```

```
 0.3311706E-08 0.1479958E-05-0.4985691E-05

 0.1566623E-09-0.3150754E-10 0.5729318E-09 0.1038621E-08-0.4253411E-08
-0.1320008E-07-0.4038229E-05 0.1337189E-04

 0.9782895E-09 0.1228656E-08 0.1038621E-08 0.2041692E-07 0.4511387E-07
 0.1426529E-06 0.3222162E-04-0.1074953E-03

 0.9452315E-09 0.5207403E-08-0.4253411E-08 0.4511387E-07 0.2394786E-06
 0.5069144E-06 0.1393375E-03-0.4637291E-03

 0.2142694E-08 0.3311706E-08-0.1320008E-07 0.1426529E-06 0.5069144E-06
 0.2970239E-05 0.7838473E-03-0.2612807E-02

 0.3587019E-06 0.1479958E-05-0.4038229E-05 0.3222162E-04 0.1393375E-03
 0.7838473E-03 0.2235445E+00-0.7462317E+00

-0.1220101E-05-0.4985691E-05 0.1337189E-04-0.1074953E-03-0.4637291E-03
-0.2612807E-02-0.7462317E+00 0.2491207E+01


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  0.3292138256471854      9.4314746735060795E-07
-1.2943698279199722E-05  6.3898140858532904E-06
TIME, RES =  0.3297096041708932      9.8383889319056395E-07
-1.3462169187739015E-05  6.6381631998145563E-06
TIME, RES =  0.3302053826946011      1.0254821639389677E-06
-1.3990683136455573E-05  6.8907373141097877E-06
TIME, RES =  0.3307011612183089      1.0680823350310564E-06
-1.4529215644931703E-05  7.1475083086563335E-06

 SIGMA IN-TRACK =  0.1698133676908E-04


 ITERATION   1

 STATE CORRECTIONS
  0.27972112873E-06 0.11856211684E-05 0.23853424221E-05 0.21854126474E-04
  0.94742438272E-04 0.20103919090E-03-0.75466819814E-02 0.28044354669E-01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12963604088E+00-0.56402464814E+00 0.86102694857E+00 0.38948223690E-01
  0.16945776838E+00 0.31689061533E+00 0.38063844048E+00 0.25505110851E+01

TIME, RES =  0.3292138256471854     -1.4932671453526947E-09
 2.3197488274728784E-07  1.4615210542322232E-07
TIME, RES =  0.3297096041708932     -7.6457434827759130E-10
 2.1318060261510396E-07  1.4535152987639466E-07
TIME, RES =  0.3302053826946011      1.1339767978790771E-09
 1.7785563821703931E-07  1.5167300177441889E-07
```

```
TIME, RES =  0.3307011612183089      4.2074803016656226E-09
 1.2606740895648727E-07  1.6505748414059884E-07

 ITERATION   2

 STATE CORRECTIONS
 -0.82168866266E-07-0.34110283260E-06-0.99106226545E-07 0.85924030800E-05
  0.37625698392E-04 0.31197374905E-04-0.19658100071E-01 0.65118972776E-01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12963612305E+00-0.56402498924E+00 0.86102684947E+00 0.38956816093E-01
  0.16949539408E+00 0.31692181271E+00 0.36098034041E+00 0.26156300579E+01


 LAST PASS RESIDUALS:

TIME, RES =  0.3292138256471854      3.5976612109550610E-08
-1.5260417348672561E-06 -5.7668950975350852E-07
TIME, RES =  0.3297096041708932      3.3442340712647178E-08
-1.4513473045596470E-06 -5.6212877703500086E-07
TIME, RES =  0.3302053826946011      3.3912736095015328E-08
-1.4196123973020391E-06 -5.2894433401513841E-07
TIME, RES =  0.3307011612183089      3.7401526525177031E-08
-1.4306708795430723E-06 -4.7728581300220119E-07

 CONVERGENCE ACHIEVED.
 IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
 DECLARIUM EST ESTIMATIA


 COVARIANCE MATRIX AT EPOCH IS:
 0.1102327E-09 0.1219909E-09 0.1826780E-09 0.1045043E-08 0.7809590E-09
 0.2183810E-08 0.3359626E-06-0.1093054E-05

 0.1219909E-09 0.5189815E-09-0.4896147E-10 0.1148637E-08 0.4579112E-08
 0.4058759E-08 0.1420575E-05-0.4582128E-05

 0.1826780E-09-0.4896147E-10 0.5692690E-09 0.1257320E-08-0.3213920E-08
-0.9928116E-08-0.2504129E-05 0.7953704E-05

 0.1045043E-08 0.1148637E-08 0.1257320E-08 0.1497127E-07 0.2603512E-07
 0.8166362E-07 0.1355449E-04-0.4332920E-04

 0.7809590E-09 0.4579112E-08-0.3213920E-08 0.2603512E-07 0.1472820E-06
 0.2734220E-06 0.5856985E-04-0.1865572E-03

 0.2183810E-08 0.4058759E-08-0.9928116E-08 0.8166362E-07 0.2734220E-06
 0.2177873E-05 0.4662139E-03-0.1490733E-02

 0.3359626E-06 0.1420575E-05-0.2504129E-05 0.1355449E-04 0.5856985E-04
 0.4662139E-03 0.1087214E+00-0.3482129E+00

-0.1093054E-05-0.4582128E-05 0.7953704E-05-0.4332920E-04-0.1865572E-03
```

```
-0.1490733E-02-0.3482129E+00 0.1115322E+01


BEGIN NEXT BAYES LOOP

 FIRST PASS RESIDUALS:

TIME, RES =  0.3540027518325782      3.8237004408234188E-06
-4.6762177792314397E-05  2.0033971476651757E-05
TIME, RES =  0.3544985303562861      3.9851018575828845E-06
-4.8597633466096468E-05  2.0796713370265646E-05
TIME, RES =  0.3549943088799939      4.1499881605275291E-06
-5.0465553836698263E-05  2.1570634670231570E-05
TIME, RES =  0.3554900874037018      4.3183636609274434E-06
-5.2365679809152077E-05  2.2355588841723589E-05

 SIGMA IN-TRACK =  0.1702700949043E-04


 ITERATION   1

 STATE CORRECTIONS
  0.11155019366E-05 0.48079306520E-05 0.84404828391E-05 0.84946458285E-04
  0.36910375836E-03 0.71775738745E-03-0.16750167522E-01 0.62508721764E-01

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12858446292E+00-0.55944941650E+00 0.86903922680E+00 0.46011654338E-01
  0.20018941065E+00 0.32992905303E+00 0.34423017288E+00 0.26781387797E+01

TIME, RES =  0.3540027518325782     -1.0538139388893697E-09
 1.1733473923580817E-06  1.0304824417627090E-06
TIME, RES =  0.3544985303562861     -1.3554804639825946E-09
 1.1330983401558470E-06  9.9100732980222928E-07
TIME, RES =  0.3549943088799939      2.3037559256089413E-09
 1.0422633877071874E-06  9.6956349003238240E-07
TIME, RES =  0.3554900874037018      9.9262575867231639E-09
 9.0123382978646305E-07  9.6592378882182084E-07

 ITERATION   2

 STATE CORRECTIONS
 -0.30380210858E-06-0.12720010757E-05-0.32951563698E-06 0.34266811185E-04
  0.14974583161E-03 0.11659816859E-03-0.52112881354E-01 0.16795677124E+00

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12858476673E+00-0.55945068850E+00 0.86903889728E+00 0.46045921149E-01
  0.20033915648E+00 0.33004565120E+00 0.29211729153E+00 0.28460955509E+01

TIME, RES =  0.3540027518325782      1.6746894476032947E-07
-5.0830708624438614E-06 -1.8168623209247059E-06
TIME, RES =  0.3544985303562861      1.5290666578737788E-07
-4.7781481982145380E-06 -1.7856776209392022E-06
TIME, RES =  0.3549943088799939      1.5050085849788175E-07
```

```
-4.6300409464183900E-06 -1.6981500181346976E-06
TIME, RES =  0.3554900874037018      1.6029546524853888E-07
-4.6380109791632762E-06 -1.5548145720113704E-06

 ITERATION   3

 STATE CORRECTIONS
  0.11027887499E-05 0.46573925932E-05-0.30154160647E-05-0.52401835723E-04
 -0.22961341017E-03 0.38702826215E-03 0.22374302593E+00-0.70853775081E+00

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12858366394E+00-0.55944603110E+00 0.86903588187E+00 0.45993519314E-01
  0.20010954307E+00 0.33043267946E+00 0.51586031746E+00 0.21375578001E+01

TIME, RES =  0.3540027518325782      1.0888437498787762E-06
 7.3654310890347574E-06  2.4609694936051502E-05
TIME, RES =  0.3544985303562861      9.7812608919334298E-07
 8.0167501843586564E-06  2.3363640254854545E-05
TIME, RES =  0.3549943088799939      8.6331014237037151E-07
 8.7128614700837659E-06  2.2100587845085701E-05
TIME, RES =  0.3554900874037018      7.4442850427069995E-07
 9.4529556180500407E-06  2.0820903901052484E-05

 ITERATION   4

 STATE CORRECTIONS
  0.17854296430E-06 0.68933950704E-06 0.37699460971E-05-0.75230045922E-04
 -0.32858211961E-03-0.71744152115E-03-0.28857294861E-02-0.97102354481E-03

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12858348539E+00-0.55944534176E+00 0.86903965181E+00 0.45918289268E-01
  0.19978096095E+00 0.32971523794E+00 0.51297458797E+00 0.21365867765E+01

TIME, RES =  0.3540027518325782     -3.0560709356544646E-07
 1.9661887642685727E-05  1.2539695299733367E-05
TIME, RES =  0.3544985303562861     -2.6422762321870463E-07
 1.8525041924455365E-05  1.2205944158995286E-05
TIME, RES =  0.3549943088799939     -2.2529963636733208E-07
 1.7423861189858059E-05  1.1858214805635092E-05
TIME, RES =  0.3554900874037018     -1.8878215958528788E-07
 1.6357548388801213E-05  1.1496854991552979E-05

 ITERATION   5

 STATE CORRECTIONS
  0.15553053302E-08 0.56803735475E-08 0.38406773695E-08-0.16855692767E-06
 -0.74431832884E-06-0.83759312294E-06 0.19481566539E-03-0.63362450061E-03

 CURRENT REFERENCE TRAJECTORY STATE VECTOR AT EPOCH:
 -0.12858348384E+00-0.55944533608E+00 0.86903965565E+00 0.45918120711E-01
  0.19978021663E+00 0.32971440034E+00 0.51316940364E+00 0.21359531520E+01
```

LAST PASS RESIDUALS:

```
TIME, RES =  0.3540027518325782      -3.0705709947245752E-07
 1.9696267535285195E-05  1.2543279857120393E-05
TIME, RES =  0.3544985303562861      -2.6548113169974630E-07
 1.8556387777368677E-05  1.2210041075121081E-05
TIME, RES =  0.3549943088799939      -2.2634733563495213E-07
 1.7452070571910561E-05  1.1862859188770114E-05
TIME, RES =  0.3554900874037018      -1.8961456738769700E-07
 1.6382517613378056E-05  1.1502082224362775E-05
```

CONVERGENCE ACHIEVED.
IN NOMINIA GAUSSIAM TRAJECTORUM REFERENTIA
DECLARIUM EST ESTIMATIA

COVARIANCE MATRIX AT EPOCH IS:

```
 0.1393081E-09 0.1337093E-09 0.2154270E-09 0.1122440E-08 0.5803153E-09
 0.8654768E-09 0.1128700E-07-0.4592820E-07

 0.1337093E-09 0.5314678E-09-0.3615734E-10 0.1045351E-08 0.3855200E-08
-0.1046738E-08 0.5760669E-07-0.1918311E-06

 0.2154270E-09-0.3615734E-10 0.5812739E-09 0.4992800E-09-0.6541382E-08
-0.1167997E-07-0.6695660E-06 0.1917969E-05

 0.1122440E-08 0.1045351E-08 0.4992800E-09 0.2982616E-07 0.9261183E-07
 0.2303867E-06 0.1126963E-04-0.3291302E-04

 0.5803153E-09 0.3855200E-08-0.6541382E-08 0.9261183E-07 0.4325458E-06
 0.9416867E-06 0.4916842E-04-0.1431688E-03

 0.8654768E-09-0.1046738E-08-0.1167997E-07 0.2303867E-06 0.9416867E-06
 0.2644483E-05 0.1335936E-03-0.3894283E-03

 0.1128700E-07 0.5760669E-07-0.6695660E-06 0.1126963E-04 0.4916842E-04
 0.1335936E-03 0.7137864E-02-0.2088942E-01

-0.4592820E-07-0.1918311E-06 0.1917969E-05-0.3291302E-04-0.1431688E-03
-0.3894283E-03-0.2088942E-01 0.6116383E-01
```

BEGIN NEXT BAYES LOOP

FIRST PASS RESIDUALS:

```
TIME, RES =  0.3787916780179711      -1.8083123288006875E-07
-1.0493742894412961E-05 -1.4617415061105032E-05
TIME, RES =  0.3792874565416789      -1.9064246669156515E-07
-1.0928857187009200E-05 -1.5242471579000189E-05
TIME, RES =  0.3797832350653868      -1.9951272854992763E-07
-1.1374361150950918E-05 -1.5865443922489370E-05
TIME, RES =  0.3802790135890946      -2.0734194169230724E-07
```

VITA

Captain Ronald A. Worley [redacted] graduated from Elyria High School in 1969. In December of 1970 he enlisted in the Air Force. In 1980 he received his Bachelor of Science in Aerospace Engineering and a commission in the USAF through AECP. His first assignment was as the project officer for Inter-Range Operations Branch, Titan Satellite Programs Division at Vandenberg AFB, CA. In May of 1984 he entered the School of Engineering, Air Force Institute of Technology.

[redacted]

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD-A164021

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED

1b. RESTRICTIVE MARKINGS:

2a. SECURITY CLASSIFICATION AUTHORITY:

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE:

3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for Public Release; Distribution Unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S): AFIT/GA/AA/85D-10

5. MONITORING ORGANIZATION REPORT NUMBER(S):

6a. NAME OF PERFORMING ORGANIZATION: School of Engineering

6b. OFFICE SYMBOL (If applicable): AFIT/AA

7a. NAME OF MONITORING ORGANIZATION:

6c. ADDRESS (City, State and ZIP Code): Air Force Institute of Technology, Wright Patterson AFB, Ohio 45433

7b. ADDRESS (City, State and ZIP Code):

8a. NAME OF FUNDING/SPONSORING ORGANIZATION:

8b. OFFICE SYMBOL (If applicable):

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER:

8c. ADDRESS (City, State and ZIP Code):

10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.

11. TITLE (Include Security Classification): See Box 19

12. PERSONAL AUTHOR(S): Ronald A. Worley, B.S., Capt., USAF

13a. TYPE OF REPORT: MS Thesis

13b. TIME COVERED: FROM ________ TO ________

14. DATE OF REPORT (Yr., Mo., Day): 1985 December

15. PAGE COUNT: 159

16. SUPPLEMENTARY NOTATION:

17. COSATI CODES

FIELD	GROUP	SUB. GR.
19	04	
16	02	

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number):
Estimation; Bayes Filter; Rocket and Missile Trajectory; Parameter Identification

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Title: ESTIMATION OF ICBM PERFORMANCE PARAMETERS

Thesis Chairman: Dr. William E. Wiesel

Approved for public release: IAW AFR 190-17.
Lynn E. Wolaver 16 JAN 86
LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (AFIC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL: Dr. William E. Wiesel, Professor

22b. TELEPHONE NUMBER (Include Area Code): 513-255-3517

22c. OFFICE SYMBOL: AFIT/ENY

DD FORM 1473, 83 APR — EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

The estimation of launch vehicle performance parameters was explored through the use of a Bayes Filter. The main emphasis was to devise the means to detect a staging event, estimate the staging time and next stage vehicle parameters, and reenter the main Bayes Filter to process subsequent stage observation data. The state model consisted of the vehicle position and velocity vectors, the exhaust velocity, and the mass ratio. The results indicated that the staging event could successfully be detected by comparing the position of the vehicle as represented by the observation data and the position as represented by the numerical integrator. The exhaust velocity and mass ratio of the next stage could not be estimated independently. The staging estimater state model was then altered to estimate the product of the exhaust velocity and mass ratio. The problems encountered reentering the main Bayes Filter were identical to the ones the staging estimator had. It was then determined that there was a possible observability problem with the main algorithm. It was recommended that the main state vector be altered to include the product of the exhaust velocity and mass ratio rather than thier independent estimation.